

Graduado en Ingeniería Informática

Universidad Politécnica de Madrid

Facultad de Informática

TRABAJO FIN DE GRADO

**DESARROLLO DE UNA HERRAMIENTA DE
GESTIÓN DE PROYECTOS Y RECURSOS EN
LA NUBE INTEGRADA EN SALESFORCE CRM**

Autor: Francisco de Borja González Orduna

Director: Felipe Fernández Hernández

MADRID, JUNIO DE 2014

Tabla de contenido

RESUMEN	vi
ABSTRACT	vii
1. INTRODUCCIÓN.....	1
1.1 HISTORIA DE LA GESTIÓN DE PROYECTOS	2
1.2 DEFINICIÓN DEL PROBLEMA	4
1.3 ALCANCE Y OBJETIVOS	5
2. ESTADO DEL ARTE	6
2.1 METODOLOGÍAS DE PLANIFICACIÓN.....	7
2.1.1 PMBOK	7
2.1.2 PRINCE2	7
2.1.3 ITIL	9
2.1.4 GESTIÓN DEL VALOR GANADO.....	9
2.1.5 GESTIÓN DE PROYECTOS POR CADENA CRÍTICA (CCPM).....	10
2.1.6 SCRUM.....	11
2.2 TECNOLOGÍAS Y HERRAMIENTAS	12
2.2.1 MICROSOFT PROJECT.....	12
2.2.2 BASECAMP	13
2.2.3 SMARTSHEET	13
2.2.4 OPENPROJECT	14
3. DESARROLLO DE LA HERRAMIENTA	15
3.1 SALESFORCE	16
3.2 MODELO DE DATOS.....	19
3.2.1 INTRODUCCIÓN	19
3.2.2 MÓDULO DE PLANIFICACIÓN DE PROYECTOS	21
3.2.2.1 PROYECTO.....	22
3.2.2.2 OBJETIVO.....	23
3.2.2.3 BLOQUE DE TAREAS.....	24
3.2.2.4 TAREA	25

3.2.2.5	PRECEDENCIA DE TAREAS	26
3.2.2.6	EMPLEADO	27
3.2.3	MÓDULO DE CALENDARIOS Y EXCEPCIONES.....	29
3.2.3.1	EMPLEADO	30
3.2.3.2	CALENDARIO	30
3.2.3.3	EXCEPCIÓN DE CALENDARIO	32
3.2.4	MÓDULO DE INFORME DE ACTIVIDADES DE TRABAJO DE EMPLEADOS	33
3.2.4.1	PROYECTO Y BLOQUE DE TAREAS.....	34
3.2.4.2	TAREA	35
3.2.4.3	ASIGNACIÓN DE TAREA	36
3.2.4.4	EMPLEADO	37
3.3	INTERFAZ DE USUARIO	39
3.3.1	INTERFAZ ESTÁNDAR	39
3.3.1.1	HOME	41
3.3.1.2	ACCOUNTS	42
3.3.1.3	PROJECTS.....	44
3.3.1.4	CALENDARS	47
3.3.1.5	EMPLOYEES	50
3.3.1.6	TASK BLOCKS	54
3.3.1.7	TASKS	55
3.3.1.8	TASK PRECEDENCES	56
3.3.1.9	TASK ASSIGNMENTS	57
3.3.1.10	CALENDAR EXCEPTIONS	58
3.3.1.11	WORK ACTIVITIES	59
3.3.2	ANÁLISIS Y GESTIÓN DE PROYECTOS.....	60
3.3.2.1	CABECERA DE PROYECTO	61
3.3.2.2	ESTADO DE COMPLETITUD Y EVM.....	62
3.3.2.3	DIAGRAMA DE GANTT	64
3.3.2.4	VISTA DE CALENDARIO.....	65
3.3.2.5	LISTADO DE TAREAS.....	68

3.3.2.6	RECURSOS HUMANOS INVOLUCRADOS	71
3.3.3	INFORME DE ACTIVIDADES DE TRABAJO	73
3.3.4	ANÁLISIS DE CARGAS DE TRABAJO DE EMPLEADOS	77
3.4	LÓGICA INTERNA PROGRAMADA	79
3.4.1	DISPARADORES	80
3.4.2	CONTROLADORES DE PÁGINAS	82
3.4.3	CLASE DE UTILIDADES	86
4.	PRUEBAS CON USUARIOS	87
5.	RESULTADOS Y CONCLUSIONES	90
ANEXO 1. CUESTIONARIO DE USABILIDAD		93
ANEXO 2. CÓDIGO		95
BIBLIOGRAFÍA		108

RESUMEN

Español

La actividad de muchas empresas, y en concreto las de *software*, está basada en proyectos. Típicamente, estas empresas tendrán un modelo de negocio orientado a productos para un mercado abierto, o un modelo de negocio orientado a consultoría para otras empresas.

La empresa LeadClic Solutions se incluye en el segundo grupo. Se trata de una consultoría especializada en una tecnología emergente, Salesforce, cuya labor consiste en el desarrollo de aplicaciones *cloud* en esta plataforma que satisfagan las necesidades de sus clientes, desde empresas relativamente pequeñas hasta grandes empresas internacionales.

Ante esta variedad de realidades, LeadClic necesita una herramienta versátil que permita una gestión eficaz de proyectos de distinta índole, integrando al mismo tiempo gestión de clientes, planificación, control de tiempo y costes, seguimiento, y gestión de recursos humanos.

En la actualidad, la empresa basa esta gestión en un gran elenco de programas especializados, desatacando hojas de cálculo, herramientas de gestión de proyectos en diversos programas locales o en la nube, correo electrónico, servicios de comunicación y desarrollos personalizados en Salesforce.

El objetivo de este Trabajo Fin de Grado es el diseño, implementación y validación de una herramienta en la nube, sobre la plataforma Salesforce, que integre las soluciones a todas las necesidades de la empresa.

ABSTRACT

English

Many companies' activity, in particular software companies', is based on projects. Typically, these companies have a product for open market - oriented business model, or a B2B consultancy business model.

The company LeadClic Solutions is included in the second group. It is a small consultancy, specialized in an emergent technology, Salesforce, that develops cloud applications in that platform to satisfy its clients' needs, from small companies to big international ones.

Faced with this variety of situations, LeadClic needs a versatile tool able to cope with an effective management of projects of different types, integrating at once client management, planning, time and costs control, monitoring, and human resources management.

At the moment, the company bases this management on a wide range of specialized programs, such as spreadsheets, local and cloud based project management tools, emailing, communication services and Salesforce based custom developments.

This Final Degree Project seeks to design, implement and validate a cloud based tool, on the Salesforce platform, to integrate the solution to all of the company's needs into one single application.

1.INTRODUCCIÓN

1.1 HISTORIA DE LA GESTIÓN DE PROYECTOS

Desde los inicios de la civilización, el hombre ha satisfecho sus necesidades mediante la realización de planes de menor o mayor complejidad. Esta capacidad le ha permitido alcanzar objetivos impensables en cada época, como la construcción de las pirámides egipcias, las obras de ingeniería romanas o la llegada del hombre a la Luna.

La sofisticación de las estructuras formales de organización humana ha crecido de manera paralela, comenzando a desarrollarse alrededor de administraciones militares, civiles y religiosas. Hasta finales del siglo XIX la gestión de proyectos era una labor principalmente creativa, pero, ya en la primera mitad del siglo XX, la industrialización dio lugar a figuras tan importantes como Henry Ford y Henry Gantt, y a herramientas de planificación tan importantes aún a día de hoy como las redes de ruta crítica o el diagrama de Gantt.

En la segunda mitad del siglo XX, la aparición de las costosas computadoras centrales (*mainframes*) permitió la utilización práctica de métodos de planificación ideados con anterioridad. La evolución de las tecnologías de la información acabó por dar lugar al escenario actual, en el que los gestores pueden hacer un uso directo de las herramientas informáticas que necesitan sin necesidad de técnicos expertos como intermediarios [1].

La popularización de Internet en los años 90 supuso una revolución en la manera en que el gestor de un proyecto y los distintos integrantes de su equipo se comunican, permitiendo la transmisión casi inmediata de grandes cantidades de información.

El desarrollo y mejora de las capacidades de internet y de los centros de datos ha dado lugar a la aparición de nuevos paradigmas en la computación distribuida, como el *Grid computing*, el *Utility computing*, los modelos “*as a Service*” y, finalmente, a la computación en la nube o *Cloud computing*. Estos nuevos modelos ofrecen a las empresas y usuarios finales tiempos y costes de implantación y mantenimiento mucho menores que los tradicionales, lo que está provocando la externalización de gran parte de las utilidades que anteriormente se proveían a nivel de equipamiento local.

A día de hoy, la definición moderna de proyecto, según el PMBOK [2], es la siguiente:

“Un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único.”

1.2 DEFINICIÓN DEL PROBLEMA

La empresa LeadClic Solutions es una consultoría especializada en la plataforma Salesforce.com. Dispone actualmente de una plantilla de quince empleados con una perspectiva de crecimiento rápido, y empieza a afrontar decisiones sobre cómo debe llevarse a cabo ese crecimiento para que se produzca de una manera ordenada y eficaz.

En la empresa, la figura del gestor de proyecto no está claramente delimitada en sus funciones, y llega a participar en los siguientes procesos de negocio fundamentales:

- Relación con el cliente, gestión de oportunidades y ventas, análisis posterior de resultados
- Planificación de proyectos
- Planificación de cargas de trabajo
- Cálculo de costes y presupuestos
- Seguimiento y gestión dinámica de proyectos en activo
- Análisis de informes de actividades de los empleados
- Comunicación interna

Acorde a estos procesos, los gestores trabajan sobre múltiples herramientas diferentes en su desempeño, destacando las siguientes:

- Salesforce CRM para gestionar relaciones con clientes, oportunidades y procesos de venta, informes y análisis
- Microsoft Project para la planificación de proyectos
- Microsoft Excel para el cálculo de presupuestos y cargas de trabajo
- Smartsheet para la gestión de proyectos activos
- Force.com Sites con un portal personalizado para el informe de actividades de trabajo por parte de los empleados
- Domino de correo electrónico y Salesforce Chatter para la comunicación interna del personal de la empresa. La última apenas se utiliza

Esta heterogeneidad provoca que los procesos fundamentales de la empresa resulten ineficientes y descoordinados, interrumpe la fluidez entre los mismos y entorpece la comunicación entre los empleados. Si no es corregido, este problema será más acusado y determinante cuanto más crezca la empresa.

1.3 ALCANCE Y OBJETIVOS

La finalidad de este trabajo es el desarrollo de una herramienta de gestión de proyectos y recursos asociados que integre todos los procesos descritos en el apartado anterior. Para ello, se ha optado por Salesforce.com como plataforma de trabajo, aprovechando las funcionalidades del CRM y de la plataforma como servicio de desarrollo Force.com.

La herramienta resultante debe cumplir los objetivos descritos a continuación:

- Ofrecer una solución para la planificación de proyectos de una potencia adecuada a las necesidades de la empresa, equivalente a otras herramientas disponibles en el mercado
- Realizar de manera sencilla y automatizada el cálculo de estimaciones, presupuestos y costes finales asociados a proyectos y personal
- Permitir la relación directa entre las ventas registradas en el CRM y los proyectos presupuestados
- Ofrecer facilidades para el análisis de cargas de trabajo del personal y su gestión, en relación con calendarios de trabajo y la planificación de proyectos
- Ofrecer una solución de gestión de proyectos en activo, que permita la realización de correcciones y cambios sobre las planificaciones iniciales de manera sencilla y en base a la información obtenida sobre cargas de trabajo y duración real de las tareas
- Permitir al personal de la empresa el registro sencillo de sus actividades de trabajo, relacionando éstas con la asignación de tareas planificada
- Facilitar la comunicación entre los empleados en base a la asignación de tareas
- Exponer una interfaz de usuario coherente, sencilla y fluida
- Cumplir los requisitos de seguridad
- Garantizar los objetivos de robustez y calidad a través de una batería de pruebas automáticas y pruebas con usuario final

2. ESTADO DEL ARTE

2.1 METODLOGÍAS DE PLANIFICACIÓN

La empresa debe decidir qué metodología de gestión de proyectos es la más adecuada para cumplir sus requerimientos. Hasta el momento, ha seguido un enfoque tradicional de desarrollo en cascada, utilizando en alguna ocasión el marco de trabajo ágil Scrum. A continuación se describen brevemente las metodologías y estándares más comunes a día de hoy.

2.1.1 PMBOK

El Project Management Institute (PMI) es un organismo internacional sin ánimo de lucro compuesto por profesionales y organizaciones relacionados con la gestión de proyectos [3]. Fue fundado en 1969, y desde la primera edición en 1987 ha publicado revisiones periódicas de su “Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK)”.

La Guía del PMBOK es uno de los más reconocidos estándares mundiales de buenas prácticas de gestión de proyectos, y constituye la norma ANSI/PMI PMI FS-PMBOK-2013 [4]. Como se especifica en el propio documento, esta guía “proporciona pautas para la dirección de proyectos individuales y define conceptos relacionados con la dirección de proyectos. Describe asimismo el ciclo de vida de la dirección de proyectos y los procesos relacionados, así como el ciclo de vida del proyecto.”

La guía está dividida principalmente en dos secciones, tratando por un lado los procesos y contextos de un proyecto, y por otro las áreas de conocimientos específicos para su gestión.

2.1.2 PRINCE2

PRINCE2 (PRojects IN Controlled Environments) es un estándar *de facto* desarrollado y utilizado por el gobierno de Reino Unido [5] que ofrece una metodología estructurada de gestión de proyectos. Se define como “basado en el producto”, ya que se centra en obtener resultados concretos y no sólo en la planificación.

La metodología que describe está basada en ocho procesos de alto nivel en el desarrollo de un proyecto, que a su vez están regidos por siete componentes que deben

estar presentes para conseguir un proyecto exitoso [6]. De estos ocho procesos, seis transcurren desde el inicio del proyecto hasta que se cierra, siendo los otros dos, planificación y dirección, procesos de apoyo. PRINCE2 se centra en la implementación de un proyecto, dejando fuera ciertos elementos de la gestión integral del mismo.

En esta metodología se dan cuatro niveles de estructura de gestión del proyecto. El primero de ellos es la gestión corporativa, que no participa en el proyecto, y los otros tres niveles se componen por el equipo de gestión del proyecto.

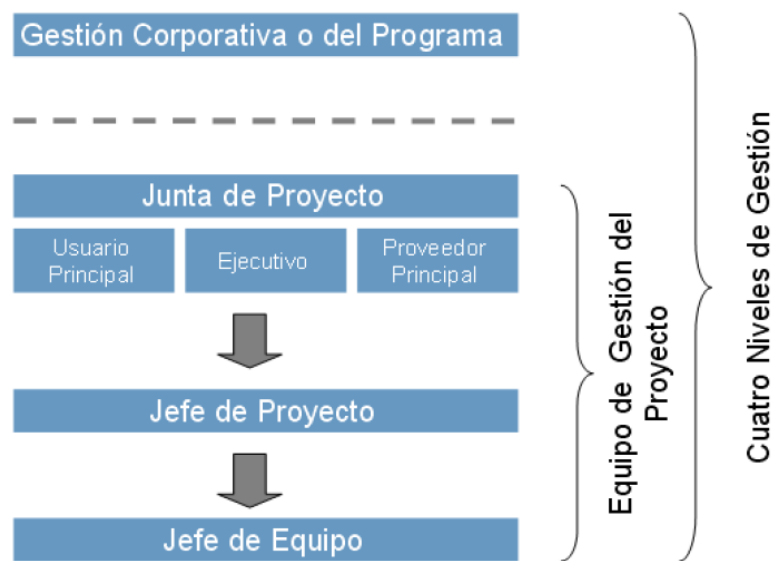


Figura 1. Estructura de gestión del proyecto

A su vez, el equipo de gestión tiene en su nivel más alto (dirección) la junta de proyecto, responsable del éxito del mismo y con capacidad de decisión. En ella participan el ejecutivo, que es el principal responsable y representa los intereses comerciales, el usuario principal, que es el responsable de la especificación y del funcionamiento del proyecto, y el proveedor principal, que representa a los desarrolladores.

El jefe de proyecto, en el nivel de gestión, se encarga de la ejecución del proyecto en su día a día y tiene responsabilidades en su producción, plazo y costes.

Por último, el jefe de equipo, en el nivel de entrega, es responsable de cada paquete de trabajo asignado por el jefe de proyecto y debe informar sobre el progreso del mismo para que los niveles superiores puedan hacer un seguimiento adecuado.

PRINCE2 asegura un ciclo de vida organizado y controlado de principio a fin, con revisiones periódicas y flexibilidad de decisiones.

2.1.3 ITIL

ITIL (Information Technology Infrastructure Library) es un estándar *de facto*, alineado con el estándar ISO 20000 [8], que proporciona un marco de trabajo de mejores prácticas en la gestión de proyectos de tecnologías de la información que tiene como el alineamiento de los servicios TI con los objetivos de negocio. Fue inicialmente escrito como una guía para el gobierno de Reino Unido en 1989 [7].

El documento está dividido principalmente en cinco grandes secciones o libros que cubren las áreas de gestión del servicio, sin llegar a ofrecer una visión holística de la gestión de proyectos:

- Estrategia: búsqueda de nuevas posibilidades de negocio
- Diseño: análisis de viabilidad, capacidades y medidas necesarias en la empresa
- Transición: validación y prueba del servicio implementado
- Operación: puesta en marcha y seguimiento del estado del servicio
- Mejora Continua: partiendo del conocimiento adquirido en la operación, se proponen e implementan soluciones y mejoras

2.1.4 GESTIÓN DEL VALOR GANADO

La metodología de gestión del valor ganado (EVM) nació en el Departamento de Defensa de Estados Unidos a finales de los 60 ante la necesidad de llevar a cabo mediciones estándar del rendimiento de los proyectos. No fue hasta principios de los 90, tras la cancelación de varios proyectos de defensa que conllevaron grandes pérdidas económicas, cuando se comenzó a valorar realmente esta metodología, llegando a la conclusión de que habrían podido prever con facilidad estos fracasos [9].

EVM es una técnica de gestión de proyectos que permite medir su rendimiento y progreso, analizando la planificación y el avance realizado para generar pronósticos de su estado futuro desde las primeras etapas del desarrollo. Las principales variables que estudia son el tiempo y el coste [10].

El correspondiente valor ganado permite controlar eficazmente la ejecución de un proyecto, evaluando el coste presupuestado del trabajo planificado, el coste presupuestado del trabajo realizado y el coste real del trabajo realizado.

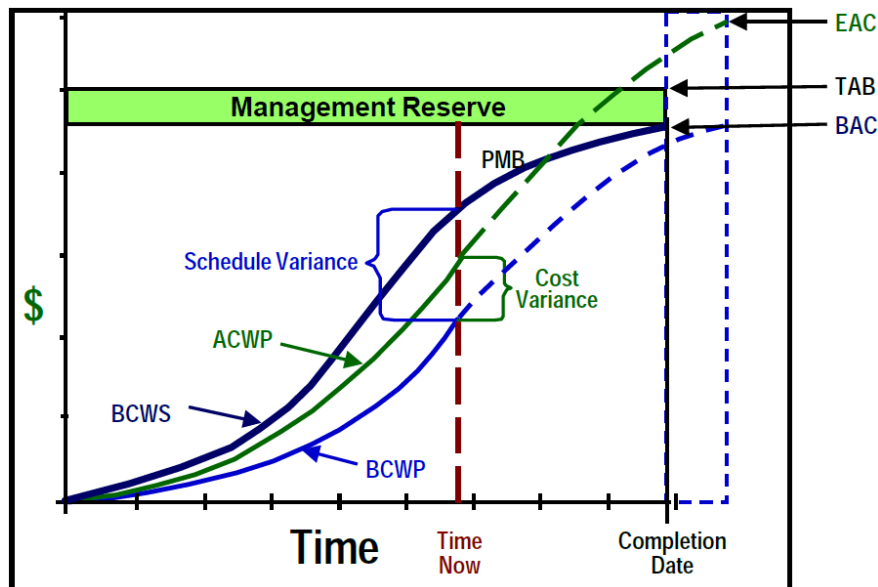


Figura 2. Gráfica de estado de un proyecto según EVM

EVM representa el valor del resultado de un proyecto en un momento dado mediante una gráfica, representando los parámetros nombrados anteriormente a lo largo del tiempo del proyecto, y realizando estimaciones sobre el futuro del proyecto basándose en la eficiencia en coste y tiempo hasta el momento [11].

2.1.5 GESTIÓN DE PROYECTOS POR CADENA CRÍTICA (CCPM)

La gestión de proyectos por cadena crítica fue introducida por Eliyahu M. Goldratt en 1997. CCPM comparte la misma base que metodologías más antiguas como CPM, en la que se calcula la ruta crítica sobre la que se subordinan el resto de tareas.

Entre sus principales características, están la identificación de dependencias implícitas en los requerimientos de recursos; la búsqueda de una solución “lo bastante buena”, ya que a pesar de ser imposible encontrar analíticamente la solución óptima, la

alcanzada mediante CCPM ofrece mucha menos incertidumbre que en estimaciones tradicionales; y la utilización de *buffers* que representan las variaciones que puede soportar la planificación [11].

Esta metodología se centra en las tareas de la cadena crítica, evitando el síndrome del estudiante, la ley de Parkinson y la mala planificación de tareas en paralelo [12].

2.1.6 SCRUM

Scrum es el marco de trabajo más común dentro de la metodología Ágil, que se contrapone a la gestión predictiva clásica. De la metodología toma su preferencia por individuos frente a procesos, *software* que funciona frente a documentación, colaboración con el cliente frente a negociación y respuesta a los cambios frente al seguimiento estricto de una planificación [13].

Según la Guía de Scrum [14]:

“Scrum se basa en la teoría de control de procesos empírica o empirismo. El empirismo asegura que el conocimiento procede de la experiencia y de tomar decisiones basándose en lo que se conoce. Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo.”

En Scrum se lleva a cabo una serie de iteraciones cortas de desarrollo o *sprints*, en la cual hay unos objetivos acordados por el equipo de desarrollo, el maestro y el dueño del producto. Además, se definen ciertos eventos, como las reuniones diarias, y artefactos, como la lista de producto.

2.2 TECNOLOGÍAS Y HERRAMIENTAS

2.2.1 MICROSOFT PROJECT

Microsoft Project es la solución ofrecida por la empresa americana para la planificación, gestión y evaluación de proyectos. Es una herramienta muy popular y potente que dispone de funcionalidades como vistas por diagrama de Gantt, gestión de recursos y sobrecargas, cálculo automático de fechas de las tareas o cálculo de la ruta crítica del proyecto. Además permite la elaboración de informes y gráficas automáticos.

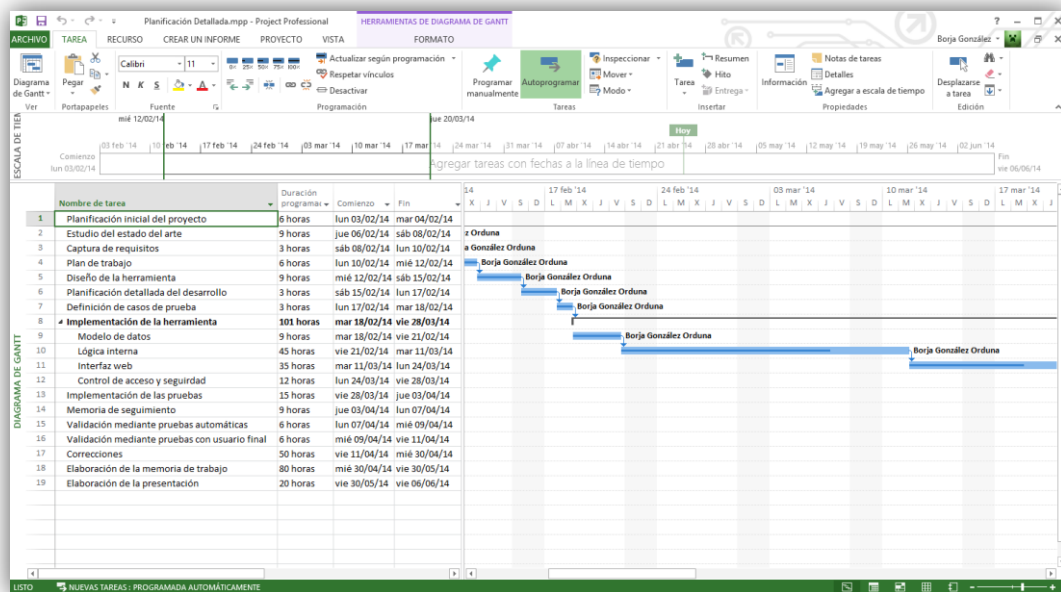


Figura 3. Interfaz de Microsoft Project 2013

2.2.2 BASECAMP

Basecamp es una herramienta SaaS¹ *web* de gestión de proyectos lanzada en 2004. Ofrece muchas de las funcionalidades clásicas de soluciones más potentes, pero con una interfaz muy sencilla y un precio relativamente reducido.

Dado que es una herramienta en la nube, aprovecha las posibilidades que ello le brinda en cuanto a una comunicación muy sencilla entre los distintos usuarios, integración con otras aplicaciones a través de su API, y su adaptabilidad a dispositivos de cualquier tipo, siempre que cuenten con un navegador *web* [15].

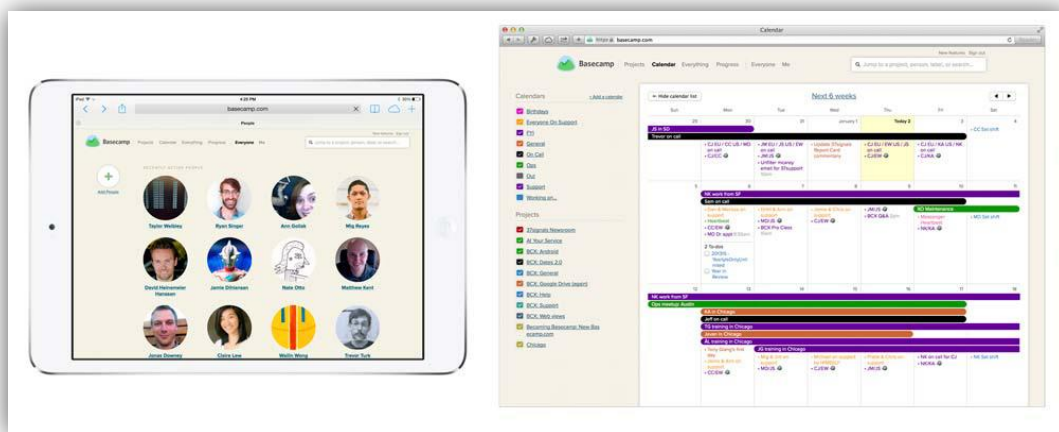


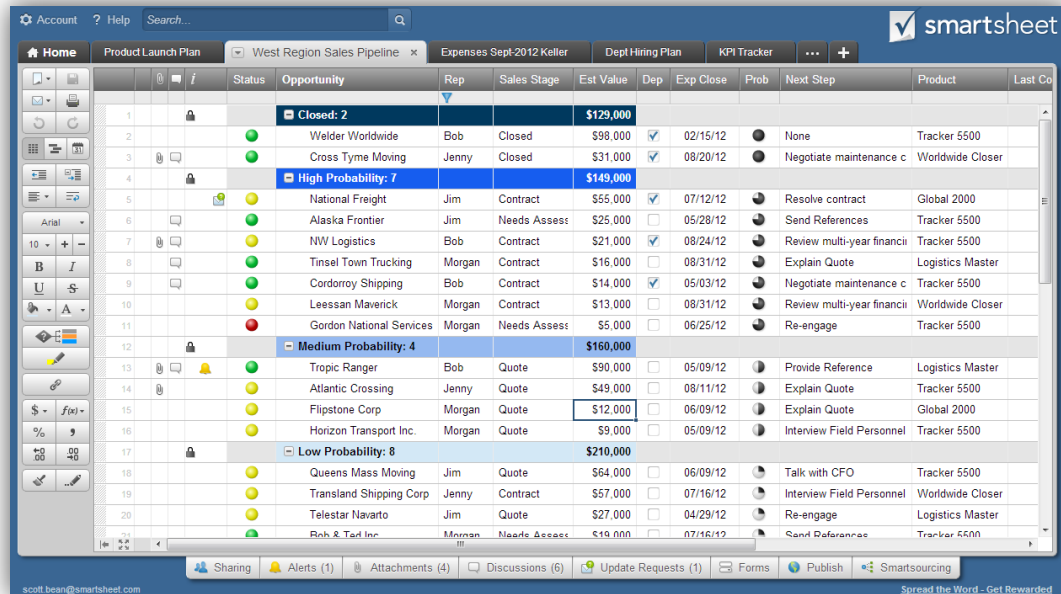
Figura 4. Interfaz web de Basecamp

2.2.3 SMARTSHEET

Siguiendo la misma premisa que Basecamp, Smartsheet es un servicio en la nube dedicado a la gestión de proyectos según el modelo SaaS. Además de las características típicas de vistas de diagrama de Gantt y gestión de tareas y recursos, Smartsheet hace hincapié en la integración con otros populares servicios, como Google Drive y Calendar, y en la comunicación rápida y eficaz entre los miembros del equipo mediante funcionalidades como notificaciones por correo electrónico del estado de las tareas, formularios *web* o transferencia de archivos [16].

¹ *Software as a Service*: Software como servicio, modelo de servicio de productos en la nube.

Smartsheet es igualmente portable a cualquier dispositivo con un navegador *web* y también dispone de API para realizar integraciones.



	Status	Opportunity	Rep	Sales Stage	Est Value	Dep	Exp Close	Prob	Next Step	Product	Last Co
1		Closed: 2			\$129,000						
2		Welder Worldwide	Bob	Closed	\$98,000		02/15/12		None	Tracker 5500	
3		Cross Tyme Moving	Jenny	Closed	\$31,000		08/20/12		Negotiate maintenance c	Worldwide Closer	
4		High Probability: 7			\$149,000						
5		National Freight	Jim	Contract	\$55,000		07/12/12		Resolve contract	Global 2000	
6		Alaska Frontier	Jim	Needs Assess	\$25,000		05/28/12		Send References	Tracker 5500	
7		NW Logistics	Bob	Contract	\$21,000		08/24/12		Review multi-year financi	Tracker 5500	
8		Tinsel Town Trucking	Morgan	Contract	\$16,000		08/31/12		Explain Quote	Logistics Master	
9		Cordoroy Shipping	Bob	Contract	\$14,000		05/03/12		Negotiate maintenance c	Tracker 5500	
10		Leessan Maverick	Morgan	Contract	\$13,000		08/31/12		Review multi-year financi	Worldwide Closer	
11		Gordon National Services	Morgan	Needs Assess	\$5,000		06/25/12		Re-engage	Tracker 5500	
12		Medium Probability: 4			\$160,000						
13		Tropic Ranger	Bob	Quote	\$90,000		05/09/12		Provide Reference	Logistics Master	
14		Atlantic Crossing	Jenny	Quote	\$49,000		08/11/12		Explain Quote	Tracker 5500	
15		Flipstone Corp	Morgan	Quote	\$12,000		06/09/12		Explain Quote	Global 2000	
16		Horizon Transport Inc.	Morgan	Quote	\$9,000		05/09/12		Interview Field Personnel	Tracker 5500	
17		Low Probability: 8			\$210,000						
18		Queens Mass Moving	Jim	Quote	\$64,000		06/09/12		Talk with CFO	Tracker 5500	
19		Transland Shipping Corp	Jenny	Contract	\$57,000		07/16/12		Interview Field Personnel	Worldwide Closer	
20		Telestar Navarto	Jim	Quote	\$27,000		04/29/12		Re-engage	Logistics Master	
21		Bob & Tadt Inc	Morgan	Needs Assess	\$19,000		07/16/12		Send References	Tracker 5500	

Figura 5. Interfaz web de Smartsheet

2.2.4 OPENPROJECT

OpenProject es una herramienta de gestión de proyectos en la nube, similar a las anteriores, pero que sigue un modelo de código abierto mediante GNU General Public License y es desarrollada por la comunidad. Fue publicada en 2010, y ofrece muchas de las funcionalidades clásicas en una herramienta de sus características.

Tiene las mismas ventajas que otras herramientas en la nube, y dispone de *plugins* para realizar reuniones y para seguir una metodología Scrum, por ejemplo.

3.DESARROLLO DE LA HERRAMIENTA

3.1 SALESFORCE

La empresa Salesforce.com Inc. fue fundada en 1999 por Marc Benioff, un antiguo directivo de Oracle, con la idea de crear *software* empresarial muy sencillo de usar y siguiendo un modelo como servicio, o SaaS, a través de internet [17].

La experiencia de Benioff en diferentes campos como el desarrollo de *software*, la gestión de varios departamentos, pero sobre todo como comercial en Oracle, le sirvió para diseñar un sistema de automatización eficiente y usable que seguía un novedoso, para su momento, sistema de pago por suscripción.

El entorno de Salesforce se ha extendiendo con el paso del tiempo y numerosas adquisiciones, integrando diferentes servicios como la gestión de ventas (Sales Cloud), la atención al cliente (Service Cloud) o la tienda de aplicaciones AppExchange, que amplían las funcionalidades de la plataforma.

Salesforce es responsable del auge del modelo *Platform as a Service*, o plataforma como servicio, que implementó en el servicio Force.com [18]. Sobre esta plataforma, desarrolladores externos pueden crear sus propias aplicaciones para satisfacer las necesidades de su empresa, clientes, o de la comunidad a través de AppExchange. Esto ha dado lugar a un extenso ecosistema y a una gran comunidad de usuarios.

Force.com sigue el patrón de diseño **Modelo-Vista-Controlador** [10], y permite la implantación de soluciones personalizadas a través de la configuración de las funcionalidades de la plataforma y la programación a través de los lenguajes Apex y Visualforce [19].

En primera instancia, al crear una nueva aplicación será necesario diseñar un **modelo** de datos relacional que represente la información que va a gestionar. Para ello pueden utilizarse objetos estándar del sistema o crear objetos personalizados a través de su interfaz *web*.

Los campos de los objetos deben ser de uno de los tipos existentes en Salesforce, incluyendo textos, números, casillas (booleanos) o dos tipos de relaciones, entre otros, y cada uno con sus propias características y parámetros a configurar. Hay muchas otras funcionalidades como reglas de validación, tipos de registro de objeto o disparadores.

Por otra parte, la **vista** es el conjunto de interfaces a través de las cuales el usuario puede obtener información del sistema, y generar cambios en el mismo. En Salesforce, el usuario accede a través de un navegador *web* a la interfaz. El desarrollador de la

aplicación puede escoger entre mostrar interfaces estándar ampliamente configurables o programar completamente páginas *web*. Esto último es posible a través del lenguaje Visualforce, un lenguaje de etiquetas muy similar a HTML, que ofrece muchos elementos coherentes con la plataforma y que es compatible con Javascript, CSS y HTML5.

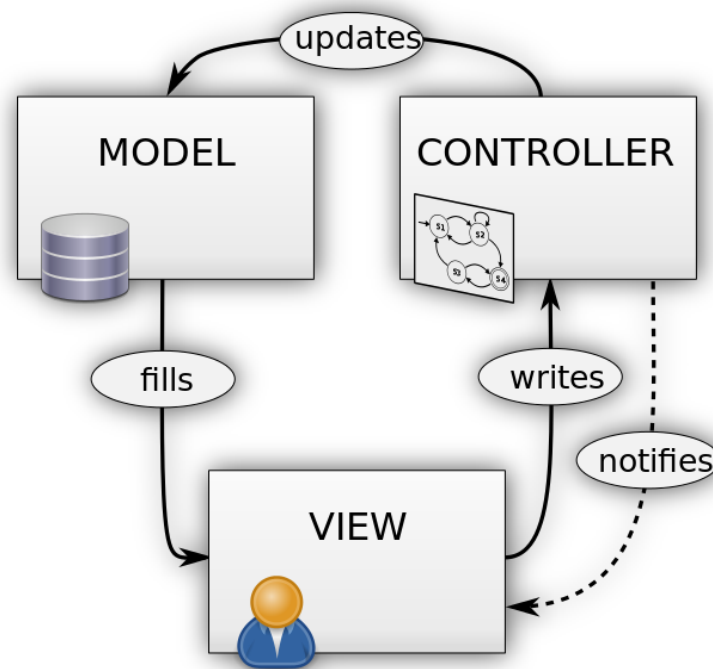


Figura 6. Modelo-Vista-Controlador

Por último, es necesario implementar la lógica (**controlador**) que regirá el modelo de datos y el procesamiento necesario para mostrar la información en las interfaces. Salesforce ofrece mucha potencia sin tener que recurrir a la programación, aprovechando sus funcionalidades configurables como flujos de trabajo, procesos de aprobación, campos sumatorios, campos fórmula o informes. Es por este motivo por el que Salesforce establece “El Fin del Software” como su misión.

Sin embargo, para los casos en que todas estas posibilidades no sean suficientemente versátiles, el desarrollador dispone de Apex, un lenguaje de programación similar a Java con llamadas integradas a la base de datos, para implementar controladores de páginas *web* y disparadores ejecutados desde la base de datos.

Es importante señalar que para evitar el consumo desmesurado de recursos en sus servidores, Salesforce impone una serie de límites que afectan casi todos los aspectos del desarrollo de una aplicación, como el número de transacciones de la base de datos, el número de relaciones entre tipos de objeto, el tiempo de procesamiento al ejecutar código o la cantidad de información que puede almacenar. Estas medidas de control influyen fundamentalmente en el diseño de las aplicaciones.



Figura 7. Logo "No Software"

3.2 MODELO DE DATOS

3.2.1 INTRODUCCIÓN

En general, el diseño de una aplicación *software* mediante el patrón Modelo-Vista-Controlador comienza con el modelo de datos de la misma, y así se ha hecho en este caso.

En este proceso se ha buscado conseguir la máxima simplicidad del modelo relacional, con el menor número de entidades (objetos de Salesforce) posible capaz de abstraer y representar toda la información que es necesario tratar. Al mismo tiempo, se ha considerado un objetivo principal el aprovechar todo lo posible las funcionalidades de Salesforce en cuanto a sus tipos de datos y relaciones entre objetos.

En Salesforce existen dos tipos de relaciones, ambas de N hijos a un padre (N:1), que asocian cada registro de un tipo de objeto (hijo), en el que se declara la relación, con un registro de tipo arbitrario (padre):

- **Campo búsqueda:** relación opcional, pudiendo almacenarse un valor nulo, y que no conlleva ningún vínculo adicional entre registros padre e hijos.
- **Campo maestro-detalle:** relación obligatoria, en la que cada registro hijo debe almacenar siempre cuál es su registro padre. Los registros hijos heredan del padre toda la información de propiedad y permisos, y son eliminados en cascada al eliminarse el primero. Además, en este tipo de relación se pueden añadir al tipo de objeto padre campos resumen automáticos de la información de sus hijos.

En este aspecto, se ha conseguido aprovechar la utilidad de las relaciones maestro-detalle y los campos resumen que proporcionan para reducir en gran medida la necesidad de programación o flujos de trabajo, con el consiguiente consumo de recursos limitados que requeriría obtener esta información de manera convencional.

Por otra parte, se ha buscado que la responsabilidad de la calidad de los datos del sistema recaiga en el propio modelo mediante reglas de validación, de manera que la programación posterior de la inteligencia sea mucho más directa. Además, el uso de campos fórmula de la base de datos ha evitado la necesidad de código para muchos de los cálculos.

En las próximas secciones se detallan el modelo² y las decisiones de diseño, separándolo en base a su área funcional en la herramienta. Dado que cada objeto puede participar en varios módulos, se describirá en cada módulo las propiedades del objeto que tengan una finalidad en él.

² En Salesforce todos los objetos incluyen un campo **Id** obligatorio, que es su identificador en el sistema, y un campo **Name** opcional, que es un nombre, así como propietario y últimas fechas de modificación y creación. Por simplicidad, no se describirán en las relaciones de campos de objetos a continuación.

3.2.2 MÓDULO DE PLANIFICACIÓN DE PROYECTOS

El primer módulo diseñado es el relativo a la planificación de proyectos. Se ha seguido un esquema jerárquico clásico de proyecto, bloque de tareas y tareas, añadiendo además un objeto que define las relaciones de precedencia entre bloques y tareas. El modelo entidad-relación, así como la descripción de los objetos, relaciones, campos y otros factores se describen a continuación.

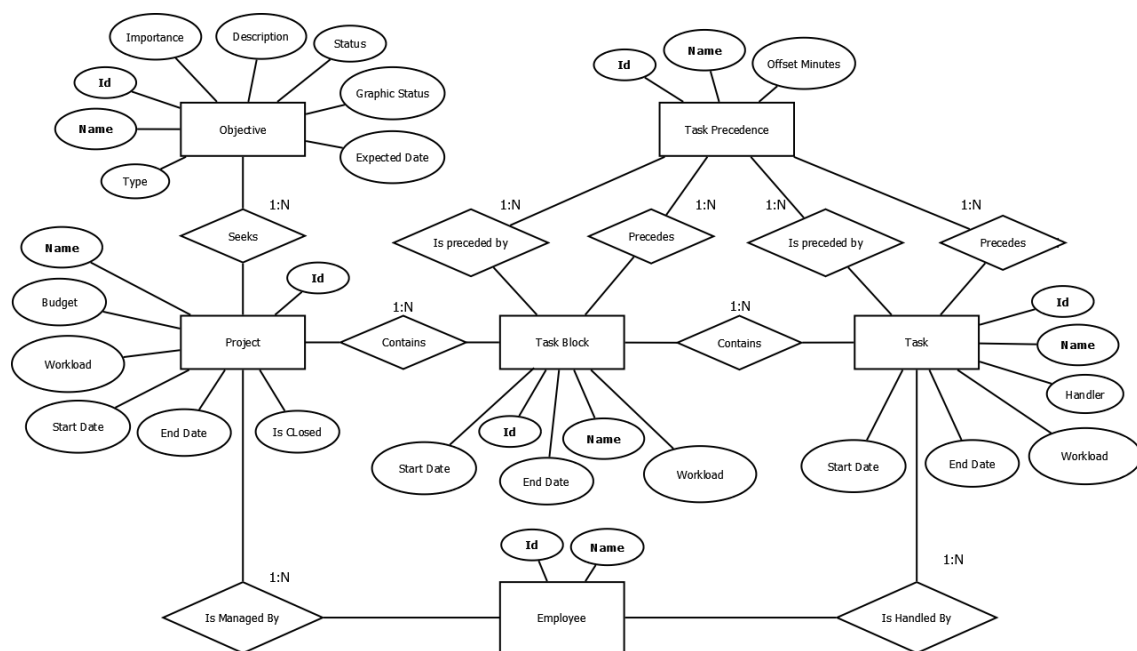


Figura 8. Módulo de planificación

3.2.2.1 PROYECTO

Descripción

Este es el objeto principal de la herramienta, ya que es la cima de la jerarquía y recopila información de los demás gracias a los campos resumen que proporcionan las relaciones maestro-detalle.

Campos

Campos personalizados:

- **Account** (fórmula, texto): enlace a la cuenta a la que pertenece la oportunidad de negocio que generó este proyecto.
- **Budget** (numérico): presupuesto planificado para el proyecto.
- **End Date** (fecha): fecha planificada de finalización del proyecto.
- **Is Closed** (booleano): define si el proyecto está abierto o cerrado.
- **Manager** (relación de búsqueda, empleado): empleado gestor del proyecto.
- **Opportunity** (relación de búsqueda, oportunidad): oportunidad de venta (CRM) que originó este proyecto.
- **Start Date** (fecha): fecha planificada de comienzo del proyecto.
- **Tasks End Date** (resumen, máximo): fecha de finalización del último bloque (y tarea) del proyecto.
- **Tasks Start Date** (resumen, mínimo): fecha de comienzo del primer bloque (y tarea) del proyecto.

Campos lista de hijos:

- **Objectives**: lista de objetivos del proyecto.
- **Task Blocks**: lista de bloques de tareas que tienen a este proyecto como padre.

Reglas de validación

- **DatesCoherence**: regla que asegura que la fecha de inicio es anterior a la de finalización.

3.2.2.2 OBJETIVO

Descripción

Este objeto representa los objetivos generales de un proyecto, sus características y su estado con indicadores gráficos.

Campos

Campos personalizados:

- **Description** (texto): descripción del objetivo.
- **Expected Date** (fecha): fecha esperada de cumplimiento del objetivo.
- **Graphic Status** (formula): imagen que representa el estado del objetivo según su tipo.
- **Importance** (lista de selección): importancia del objetivo para el éxito del proyecto.
- **Project** (relación de búsqueda, proyecto): proyecto al que pertenece este objetivo.
- **Status** (lista de selección): estado del objetivo en cuanto a su cumplimiento.
- **Type** (lista de selección): tipo del objetivo en cuanto a su obligatoriedad.

Campos lista de hijos:

- **Dependent Tasks**: lista de tareas relacionadas con este objetivo.

3.2.2.3 BLOQUE DE TAREAS

Descripción

El objetivo de este objeto es el de agrupar tareas para facilitar su gestión. Para simplificar el modelo y permitir que el resumen automático de información pueda ascender desde las tareas hasta el proyecto, se ha establecido que todas las tareas deban pertenecer siempre a un bloque de tareas, aunque sean únicas. Se puede incluir un campo adicional en el objeto bloque que indique que éste es transparente al usuario y así mostrar las tareas como si no pertenecieran a ninguno.

Campos

Campos personalizados:

- **Completion** (resumen, sumatorio): suma de las cargas de trabajo de las asignaciones de tareas del bloque multiplicadas por su estado de completitud, obtenidas a través de las tareas descendientes, medida en minutos.
- **End Date** (resumen, máximo): fecha de finalización de la última tarea del bloque.
- **Project** (relación maestro-detalle, proyecto): proyecto al que pertenece este bloque.
- **Start Date** (resumen, mínimo): fecha de comienzo de la primera tarea del bloque.

Campos lista de hijos:

- **Tasks**: lista de tareas que tienen a este bloque como padre.
- **Task Dependencies**: lista de relaciones de precedencia de tareas en las que este bloque depende de otro bloque o tarea.
- **Task Precedences**: lista de relaciones de precedencia de tareas en las que este bloque precede a otro bloque o tarea.

3.2.2.4 TAREA

Descripción

Este objeto representa cada tarea individual que se ha planificado para la realización del proyecto. Cada tarea debe indicar un objetivo mínimo en el desarrollo, y debe tener un responsable de su realización en el tiempo y forma definidos.

Campos

Campos personalizados:

- **Base End Date** (resumen, máximo): fecha de finalización calculada de la última asignación de la tarea según la carga de trabajo de los empleados.
- **Base Start Date** (resumen, mínimo): fecha de comienzo calculada de la primera asignación de la tarea según la carga de trabajo de los empleados.
- **End Date** (fecha): fecha planificada de finalización de la tarea.
- **Handler** (relación de búsqueda, empleado): empleado responsable de esta tarea.
- **Project** (fórmula,texto): enlace al proyecto al que pertenece la tarea. Es un campo informativo.
- **Related Objective** (relación de búsqueda, objetivo): objetivo del proyecto con el que está relacionada esta tarea.
- **Start Date** (fecha): fecha planificada de comienzo de la tarea.
- **Task Block** (relación maestro-detalle, bloque de tareas): bloque al que pertenece esta tarea.
- **Workload** (numérico): carga de trabajo planificada de la tarea medida en minutos.

Campos lista de hijos:

- **Task Dependencies**: lista de relaciones de precedencia de tareas en las que esta tarea depende de otro bloque o tarea.
- **Task Precedences**: lista de relaciones de precedencia de tareas en las que esta tarea precede a otro bloque o tarea.

Reglas de validación

- **DatesCoherence:** regla que asegura que la fecha de inicio es anterior a la de finalización.
- **TaskInProject:** regla que asegura que las fechas de comienzo y finalización de la tarea se encuentran dentro de las del proyecto.

3.2.2.5 PRECEDENCIA DE TAREAS

Descripción

Mediante este objeto se establecen las relaciones de orden cronológico entre bloques y tareas. Dado que Salesforce sólo permite relaciones 1:N, ha sido necesario crear este objeto intermedio que permite las relaciones de precedencia N:N y la inclusión de atributos en la relación.

Campos

Campos personalizados:

- **Dependent Block** (relación de búsqueda, bloque de tareas): en el caso de que la relación establezca un bloque dependiente, es informado en este campo.
- **Dependent Duration** (fórmula, numérico): duración en minutos de la tarea o bloque dependiente en la relación para facilitar la programación.
- **Dependent Start Date** (fórmula, fecha/hora): fecha y hora de inicio de la tarea o bloque dependiente en la relación para facilitar la programación.
- **Dependent Task** (relación de búsqueda, tarea): en el caso de que la relación establezca una tarea dependiente, es informada en este campo.
- **Offset Minutes** (numérico): desfase en minutos entre el minuto de finalización del bloque o tarea precedente y el minuto de comienzo del bloque o tarea dependiente.
- **Precedent End Date** (fórmula, fecha/hora): fecha y hora de fin de la tarea o bloque precedent en la relación para facilitar la programación.
- **Preceding Block** (relación de búsqueda, bloque de tareas): en el caso de que la relación establezca un bloque precedente, es informado en este campo.

- **Preceding Task** (relación de búsqueda, tarea): en el caso de que la relación establezca una tarea precedente, es informada en este campo.

Reglas de validación

- **NotModifiable**: regla que asegura que los bloques y tareas precedentes y dependientes no son modificados.
- **OnlyOnePrecedentDependent**: regla que asegura sólo hay un bloque o tarea precedente y dependiente.
- **RequiredPrecedentAndDependent**: regla que asegura que siempre exista un bloque o tarea precedente y dependiente.

3.2.2.6 EMPLEADO

Descripción

Este objeto contiene la información relativa a un empleado de la empresa, que en este módulo cumple las funciones de gestor de un proyecto o de responsable de una tarea.

Campos

Campos personalizados:

- **Email** (correo electrónico): dirección de correo electrónico por la que se puede contactar con el empleado.
- **Labour Cost** (moneda): coste salarial por hora del empleado.
- **Reporting Password** (texto encriptado): contraseña del empleado para acceder a la página de informe de tiempos de trabajo.
- **Salesforce User** (relación de búsqueda, usuario): en caso de que el empleado tenga una licencia de usuario de Salesforce, se informa en este campo.

Campos lista de hijos:

- **Managed Projects:** lista de proyectos de los que este empleado es gestor.
- **Handled Tasks:** lista de tareas de las que este empleado es responsable.

3.2.3 MÓDULO DE CALENDARIOS Y EXCEPCIONES

El módulo de calendarios y excepciones representa la información relativa a calendarios y horarios de trabajo, así como ausencias debidas a festivos, vacaciones u otras causas. El modelo entidad-relación, así como la descripción de los objetos, relaciones, campos y otros factores se describen a continuación.

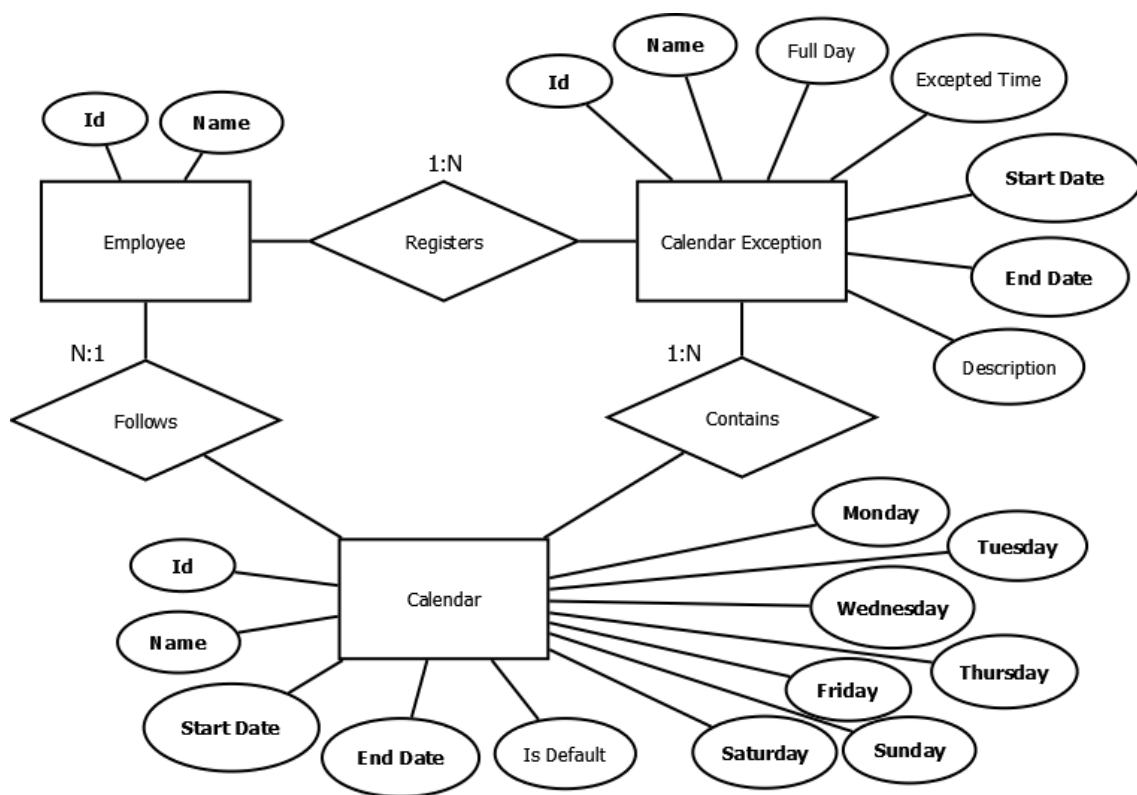


Figura 9. Modelo de calendarios y excepciones

3.2.3.1 EMPLEADO

Descripción

Hace referencia a un empleado de la empresa gestionada. En este módulo se registra el calendario que sigue el empleado y las excepciones adicionales que solicite.

Campos

Campos personalizados:

- **Calendar** (relación de búsqueda, calendario): calendario del sistema que el empleado sigue como base para establecer su horario y sus excepciones, como festivos y días libres de la empresa. Es un campo obligatorio, necesario para muchos de los cálculos efectuados en la herramienta.

Campos lista de hijos:

- **Personal Exceptions:** lista de excepciones de calendario asignadas al empleado, que pueden ser por vacaciones u otros motivos

3.2.3.2 CALENDARIO

Descripción

El objeto calendario almacena información sobre los tiempos de trabajo de cada día de la semana que tendrán los empleados subscritos al mismo, así como una lista de excepciones relacionadas que les serán aplicadas.

Campos

Campos personalizados:

- **End Date** (fecha): fecha en la que el calendario deja de tener efecto.

- **Friday** (numérico): tiempo de trabajo que tendrá un empleado los viernes, en minutos.
- **Is Default** (booleano): define si este es el calendario asignado por defecto a los empleados.
- **Monday** (numérico): tiempo de trabajo que tendrá un empleado los lunes, en minutos.
- **Saturday** (numérico): tiempo de trabajo que tendrá un empleado los sábados, en minutos.
- **Start Date** (fecha): fecha en la que el calendario empieza a tener efecto.
- **Sunday** (numérico): tiempo de trabajo que tendrá un empleado los domingos, en minutos.
- **Thursday** (numérico): tiempo de trabajo que tendrá un empleado los jueves, en minutos.
- **Tuesday** (numérico): tiempo de trabajo que tendrá un empleado los martes, en minutos.
- **Wednesday** (numérico): tiempo de trabajo que tendrá un empleado los miércoles, en minutos.

Campos lista de hijos:

- **Employees**: lista de empleados que siguen este calendario.
- **Exceptions**: lista de excepciones asignadas a este calendario, y que se aplican a todos los empleados que lo siguen.

Reglas de validación

- **DatesCoherence**: regla que asegura que la fecha de inicio es anterior a la de finalización.

3.2.3.3 EXCEPCIÓN DE CALENDARIO

Descripción

La excepción de calendario representa un día que no cumple con lo establecido por un calendario como normal. Puede ser una excepción personal de un empleado, o una excepción general de un calendario, que se aplicará a todos los empleados suscritos.

Campos

Campos personalizados:

- **Calendar** (relación de búsqueda, calendario): si esta excepción está asignada a un calendario, se informa en este campo.
- **Employee** (relación de búsqueda, empleado): si esta excepción está asignada a un empleado, se informa en este campo.
- **End Date** (fecha): fecha en la que finaliza la excepción.
- **Excepted Time** (numérico): tiempo de trabajo perdido por la excepción cada día que dure, en minutos.
- **Full Day** (booleano): indica si la excepción se aplica a los días completos en los que sea efectiva.
- **Reason** (texto): motivo por el que se aplica la excepción.
- **Start Date** (fecha): fecha en la que comienza la excepción.

Reglas de validación

- **DatesCoherence**: regla que asegura que la fecha de inicio es anterior a la de finalización.
- **ExceptionType**: regla que asegura que se especifique el tiempo de trabajo perdido si, y sólo si la excepción es del tipo de todo el día.
- **OnlyOneFather**: regla que asegura que una excepción sólo esté asignada a un empleado o a un calendario.

3.2.4 MÓDULO DE INFORME DE ACTIVIDADES DE TRABAJO DE EMPLEADOS

El módulo de informes de actividades de trabajo representa la información relativa a los informes que los empleados completan sobre las actividades que realizan a través de la página de *reporting*, permitiendo controlar la variación en tiempo y costes de las asignaciones de tareas de un proyecto. El modelo entidad-relación, así como la descripción de los objetos, relaciones, campos y otros factores se describen a continuación.

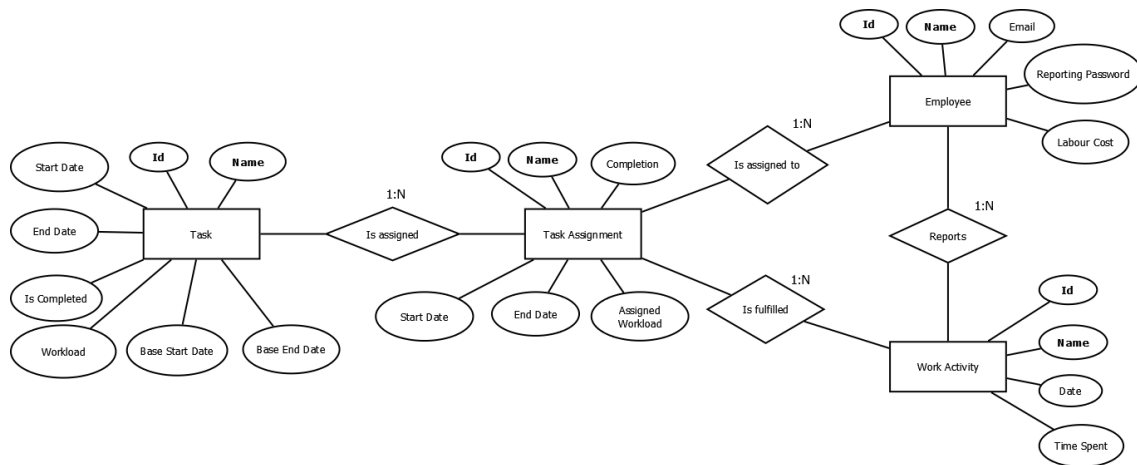


Figura 10. Modelo de informes

3.2.4.1 PROYECTO Y BLOQUE DE TAREAS

Descripción

Aunque no se hayan incluido en el modelo para este módulo, los objetos proyecto y bloque poseen algunos campos resumen de jerarquía (proyecto sobre bloque, bloque sobre tarea) que recogen información referida a la gestión e informe de actividades.

Campos

Campos personalizados:

- **Assigned Cost** (resumen, sumatorio): sumatorio del coste de las asignaciones hechas a empleados en función de los tiempos de trabajo y costes salariales.
- **Assigned Workload** (resumen, sumatorio): sumatorio de las cargas de trabajo que ya han sido asignadas a empleados. Se mide en minutos y muestra la carga de trabajo total asignada.
- **Completion** (resumen, sumatorio): sumatorio del estado de completitud de las asignaciones descendientes. Se mide en minutos, y siempre será menor o igual a la carga de trabajo asignada, para que los cálculos porcentuales sean precisos.
- **Incurred Cost** (resumen, sumatorio): suma de los costes reales de las actividades de trabajo descendientes.
- **Tasks Workload** (resumen, sumatorio): sumatorio de las cargas de trabajo planificadas de las tareas descendientes. Se mide en minutos y muestra la carga de trabajo total planificada para el proyecto.

3.2.4.2 TAREA

Descripción

Este objeto representa cada tarea individual que se ha planificado en el proyecto. Cada tarea tiene un responsable, y su realización se distribuirá entre las asignaciones que se hagan de ella a los empleados.

Campos

Campos personalizados:

- **Assigned Cost** (moneda): coste de las asignaciones asociadas a esta tarea en función de las cargas de trabajo y los costes salariales. Este cálculo se realiza programáticamente en vez de mediante campo resumen por limitación del sistema.
- **Assigned Workload** (resumen, sumatorio): sumatorio de las cargas de trabajo que ya han sido asignadas a empleados. Se mide en minutos.
- **Base End Date** (resumen, máximo): fecha en la que se terminará la tarea basándose en las fechas de finalización calculadas para las asignaciones, teniendo en cuenta las cargas de trabajo de los empleados.
- **Base Start Date** (resumen, mínimo): fecha en la que comenzará la tarea basándose en las fechas de comienzo calculadas para las asignaciones, teniendo en cuenta las cargas de trabajo de los empleados.
- **Completion** (resumen, sumatorio): sumatorio del estado de completitud de las asignaciones de la tarea. Se mide en minutos, y siempre será menor o igual a la carga de trabajo asignada, para que los cálculos porcentuales sean precisos.
- **Incurred Cost** (resumen, sumatorio): suma de los costes reales de las actividades de trabajo de la tarea, obtenidos a través de las asignaciones.
- **Is Completed** (fórmula, booleano): define si esta tarea ya se ha completado, es decir, si el campo *Completion* iguala al campo *Assigned Workload*.

Campos lista de hijos:

- **Task Assignments**: lista de asignaciones realizadas de esta tarea a empleados.

Reglas de validación

- **MaximunWorkload:** regla que asegura que no hay más carga de trabajo asignada que la planificada, para que se realicen cálculos de completitud correctos.

3.2.4.3 ASIGNACIÓN DE TAREA

Descripción

El objeto asignación de tarea relaciona una tarea, los empleados que deben completarla y las actividades de trabajo que estos realizan para la asignación.

Campos

Campos personalizados:

- **Assigned Workload** (resumen, sumatorio): carga de trabajo en minutos de la asignación.
- **Base End Date** (resumen, máximo): fecha en la que debería finalizarse la asignación, teniendo en cuenta la carga de trabajo del empleado.
- **Base Start Date** (resumen, mínimo): fecha en la que debería comenzarse la asignación, teniendo en cuenta la carga de trabajo del empleado.
- **Completion** (resumen, sumatorio): completitud en minutos de manera relativa a la carga asignada inicialmente.
- **Employee** (relación de búsqueda, empleado): empleado que debe realizar esta asignación.
- **End Date** (fecha): fecha de finalización planificada.
- **Estimated Cost** (moneda): coste de la asignación basado en la carga de trabajo asignada y el coste salarial del empleado.
- **Project** (fórmula,texto): enlace al proyecto al que pertenece la asignación. Es un campo informativo.
- **Real Cost** (moneda): coste de la asignación basado en la duración de las actividades de trabajo registradas por los empleados y sus costes salariales. Este

cálculo se realiza programáticamente en vez de mediante campo resumen por limitación del sistema.

- **Start Date** (fecha): fecha de comienzo planificada.
- **Task** (relación maestro-detalle, tarea): tarea relacionada con la asignación.

Campos lista de hijos:

- **Work Activities**: lista de actividades de trabajo realizadas por empleados relacionadas con la asignación.

Reglas de validación

- **CompletionMax**: regla que asegura que la completitud sea menor o igual que la carga de trabajo asignada, ya que la proporción es utilizada para calcular el tanto por ciento de manera precisa.

3.2.4.4 EMPLEADO

Descripción

Hace referencia a un empleado de la empresa gestionada. En este módulo se registran las asignaciones relativas a tareas que se dan a cada empleado, y las actividades de trabajo que el mismo realiza en relación a cada asignación.

Campos

Campos personalizados:

- **Email** (email, único): dirección de correo electrónico del empleado, necesaria para contacto y para acceder a la página de informe de actividades de trabajo.
- **Labour Cost** (moneda): coste salarial por hora.
- **Reporting Password** (texto, encriptado): contraseña para acceder a la página de informe de actividades de trabajo, almacenado cifrado mediante AES-128 [20].

Campos lista de hijos:

- **Task Assignments:** lista de asignaciones de tareas hechas al empleado.
- **Work Activities:** lista de actividades de trabajo registradas por el empleado.

3.3 INTERFAZ DE USUARIO

3.3.1 INTERFAZ ESTÁNDAR

La interfaz estándar de Salesforce, es decir, el conjunto de páginas *web* de la plataforma que permiten al usuario interactuar con el sistema para configurarlo, extenderlo y manipular la información que contiene, es muy potente. Podría dividirse en dos apartados:

- Uso de la herramienta, cuya estructura está dividida en pestañas para acceder a los registros de cada tipo de objeto o a páginas personalizadas
- Configuración, relativamente compleja, y navegable mediante un panel lateral con estructura de árbol.

En esta sección se van a mostrar las interfaces de uso estándar, personalizadas para cada objeto de forma que muestren la información de los registros de la manera más organizada y accesible posible.

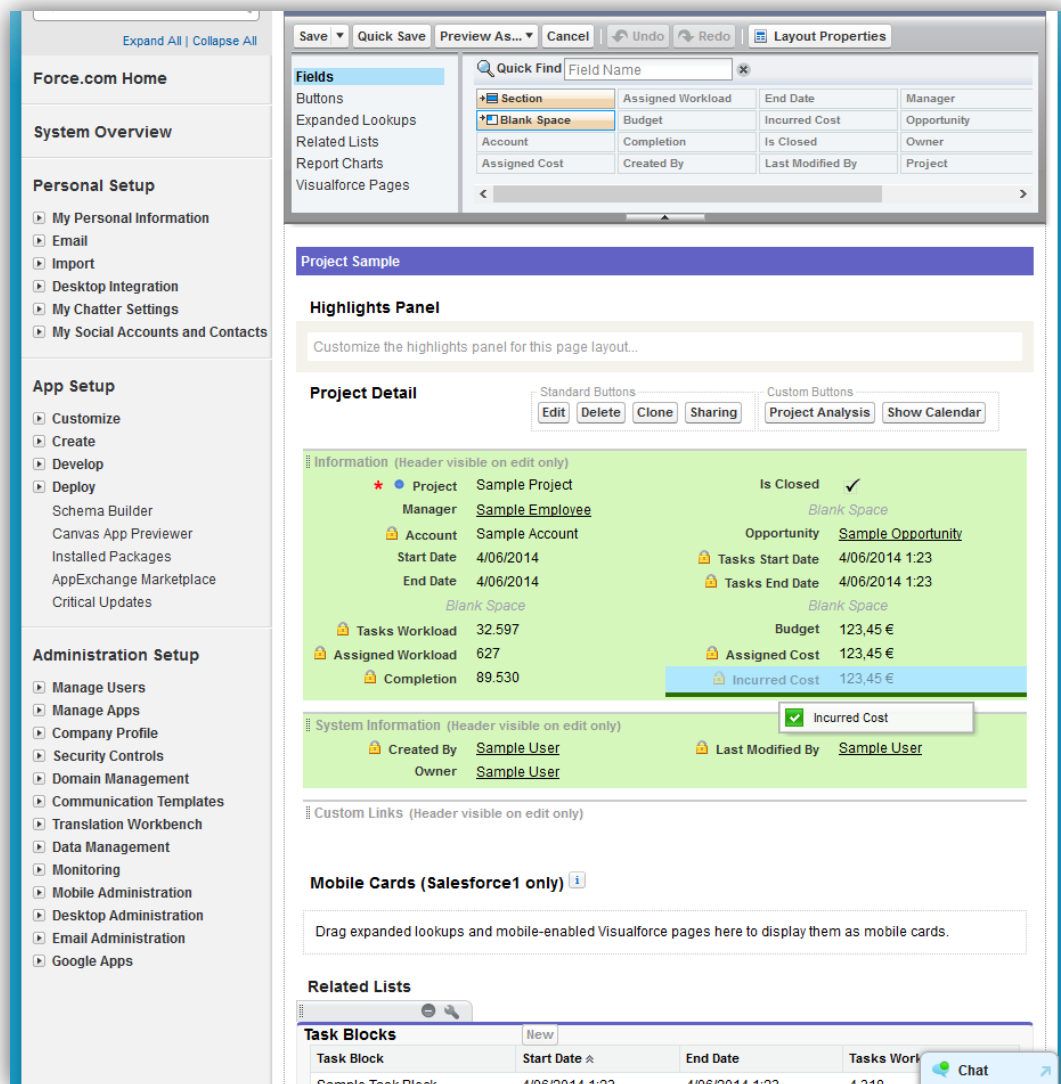


Figura 11. Ejemplo de configuración del formato de página de objeto *Project*

Para cada tipo de objeto existe una página de búsqueda con diferentes “vistas” editables, que se puede asignar a una pestaña de la interfaz principal, y una página con formato configurable para mostrar la información de los registros (Figura 11. Ejemplo de configuración del formato de página de objeto *Project*).

3.3.1.1 HOME

Esta es la página de entrada a la aplicación, asignada a la primera pestaña. Desde ella se puede acceder al resto de pestañas de la herramienta, crear nuevos registros de cualquier tipo de objeto, acceder a una papelera de registros eliminados, a la información del usuario actual y se puede observar un panel (desarrollado de manera personalizada) en el que aparece la información general de los proyectos activos actualmente.

The screenshot displays the ProMan application's Home page. At the top, there's a navigation bar with tabs: Home, Accounts, Projects, Calendars, Employees, Resources, and Workload. Below this, a user profile for Borja Gonzalez Orduna is shown, including a 'Show Feed' button and a 'Spring '14 for Developers' badge. On the left, a 'Recent Items' sidebar lists various project-related items. The main content area features a 'Current Projects' table with the following data:

Project	Start Date	End Date	Workload	Budget	incurred Cost	Completion %
Test Project 2	19/03/2014	14/05/2014	0 h	15000.00 €	0.00 €	0 %
Test Project	12/03/2014	24/05/2014	68 h	25000.00 €	258.33 €	1 %
Desarrollo de herramienta de TFG	24/03/2014	10/11/2014	350 h	5000.00 €	0.00 €	0 %

At the bottom, there's a footer with copyright information and a 'Chat' button.

Figura 12. Página Home

3.3.1.2 ACCOUNTS

Esta es la página de visualización de cuentas, asignada a la segunda pestaña. En ella se puede acceder a los registros de tipo cuenta, un objeto estándar de Salesforce que hace referencia a una entidad (organización, empresa, etc.) con la que se tiene relación.

Tras pasar la página de elementos recientes, se puede acceder a la vista del listado de cuentas.

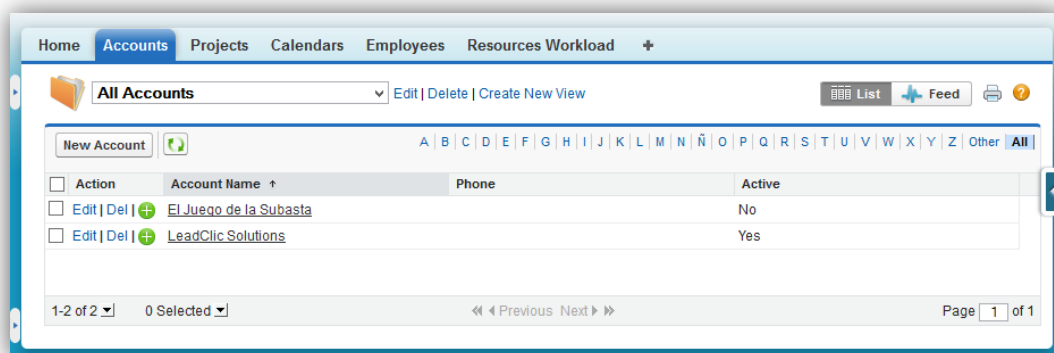


Figura 13. Pestaña *Accounts*: listado

Al pulsar en uno de los registros, se accede a toda su información según el formato de página configurado. En este formato se muestra el valor de los campos de cuenta organizados en secciones, así como las listas de contactos y oportunidades de negocio asociados.

The screenshot displays the 'LeadClic Solutions' account detail page. The navigation bar at the top includes links for Home, Accounts, Projects, Calendars, Employees, Resources, and Workload. The page title is 'LeadClic Solutions'. Below the title, there are links for 'Customize Page', 'Edit Layout', 'Printable View', and 'Help for this Page'. A 'Show Feed' button is also present. The page is organized into several sections:

- Account Detail:** Includes buttons for 'Edit', 'Delete', and 'Manage External Account'.
- Account Information:** A table with fields:

Account Name	LeadClic Solutions View Hierarchy	Rating	Warm
Parent Account		SIC Code	B12345678
Type	Technology Partner	Industry	Technology
Employees	14	Active	Yes
Description	Salesforce Consultory		
- Contact Information:** A table with fields:

Phone	913555555	Fax	
Website	http://www.leadclic.com		
- Address Information:** A table with fields:

Billing Address	Shipping Address
-----------------	------------------
- System Information:** A table with fields:

Created By	Boria Gonzalez Orduna , 26/05/2014 10:23		Last Modified By	Boria Gonzalez Orduna , 4/06/2014 2:06
Account Owner	Boria Gonzalez Orduna [Change]			
- Contacts:** A section with buttons for 'New Contact' and 'Merge Contacts'. It shows 'No records to display'.
- Opportunities:** A section with a 'New Opportunity' button. It shows a table with one record:

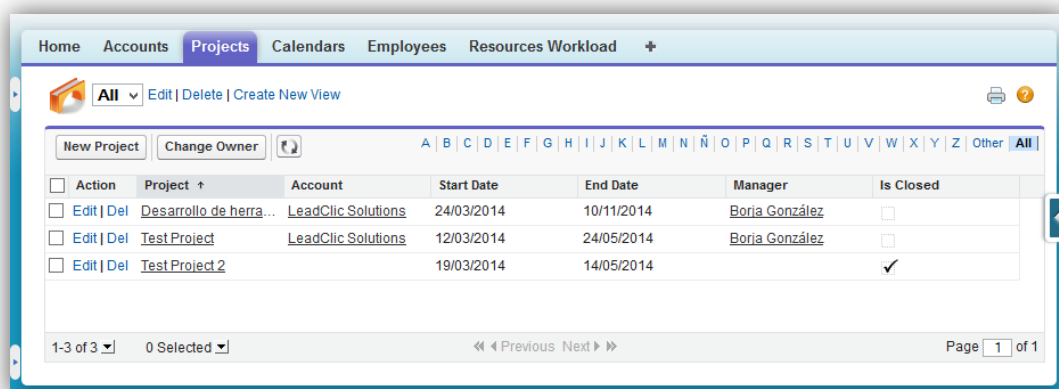
Action	Opportunity Name	Stage	Amount	Close Date
Edit Del	PM Tool	Closed Won		6/06/2014

At the bottom, there are links for 'Back To Top' and a note: 'Always show me fewer ▲ / ▼ more records per related list'.

Figura 14. Account: Formato de página

3.3.1.3 PROJECTS

Esta es la página de visualización de proyectos, asignada a la tercera pestaña. En ella se puede acceder a los registros de tipo proyecto. Tras pasar la página de elementos recientes, se puede acceder a la vista del listado de proyectos.



Action	Project	Account	Start Date	End Date	Manager	Is Closed
Edit Del	Desarrollo de herra...	LeadClic Solutions	24/03/2014	10/11/2014	Boria González	<input type="checkbox"/>
Edit Del	Test Project	LeadClic Solutions	12/03/2014	24/05/2014	Boria González	<input type="checkbox"/>
Edit Del	Test Project 2		19/03/2014	14/05/2014		<input checked="" type="checkbox"/>

Figura 15. Pestaña *Projects*: listado

Al pulsar en uno de los registros, se accede a toda su información según el formato de página configurado. En este formato se muestra el valor de los campos de proyecto divididos por secciones, así como las listas de bloques de tareas en que se divide y de objetivos.

Project
Desarrollo de herramienta de TFG

Customize Page | Edit Layout | Printable View | Help for this Page

« Back to List: Accounts

Task Blocks [5] | Objectives [5]

Project Detail [Edit] [Delete] [Clone] [Project Analysis] [Show Calendar]

▼ **Information**

Project	Desarrollo de herramienta de TFG	Account	LeadClic Solutions
Manager	Boria González	Opportunity	PM Tool
Start Date	24/03/2014	Tasks Start Date	30/03/2014 9:52
End Date	10/11/2014	Tasks End Date	8/11/2014 0:58
		Is Closed	<input type="checkbox"/>

▼ **Workload and Costs**

Tasks Workload	21.020	Budget	5.000,00 €
Assigned Workload	13.520	Assigned Cost	5.031,24 €
Completion	0	Incurred Cost	0,00 €

▼ **System Information**

Created By	Boria Gonzalez Orduna, 29/05/2014 9:21	Last Modified By	Boria Gonzalez Orduna, 3/06/2014 0:53
Owner	Boria Gonzalez Orduna [Change]		

[Edit] [Delete] [Clone] [Project Analysis] [Show Calendar]

Task Blocks [New Task Block] [Task Blocks Help]

Action	Task Block	Start Date	End Date	Tasks Workload
[Edit] [Del]	Análisis de requisitos	30/03/2014 9:52	18/04/2014 10:58	620
[Edit] [Del]	Diseño de la herramienta	18/04/2014 10:58	1/05/2014 23:57	600
[Edit] [Del]	Implementación	5/05/2014 23:57	30/09/2014 23:57	14.400
[Edit] [Del]	Validación	25/07/2014 0:57	8/11/2014 0:58	3.400
[Edit] [Del]	Incidencias y mejoras no previstas	13/09/2014 1:00	28/10/2014 1:00	2.000

Objectives [New Objective] [Objectives Help]

Action	Objective	Expected Date	Importance	Type	Status	Graphic Status
[Edit] [Del]	Cumplir fecha de entrega	6/06/2014	High	Mandatory	In Progress	
[Edit] [Del]	Cumplir presupuesto	5/06/2014	Critical	Mandatory	In Progress	
[Edit] [Del]	Definición precisa de requisitos	30/11/2013	High	Mandatory	Achieved	✓
[Edit] [Del]	Optimizar recursos		Medium	Optional	In Progress	
[Edit] [Del]	Modularizar desarrollo		Low	Idea	Failed	ⓘ

^ Back To Top

Always show me fewer ▲ / ▼ more records per related list

Figura 16. Project: Formato de página

Se han creado también dos botones personalizados, que abren sendas páginas personalizadas con una ventana emergente de una vista de calendario en la que se muestran los bloques, las tareas y las excepciones del calendario por defecto del sistema, y la interfaz de control de proyectos (ANÁLISIS Y GESTIÓN DE PROYECTOS, pg. 60).

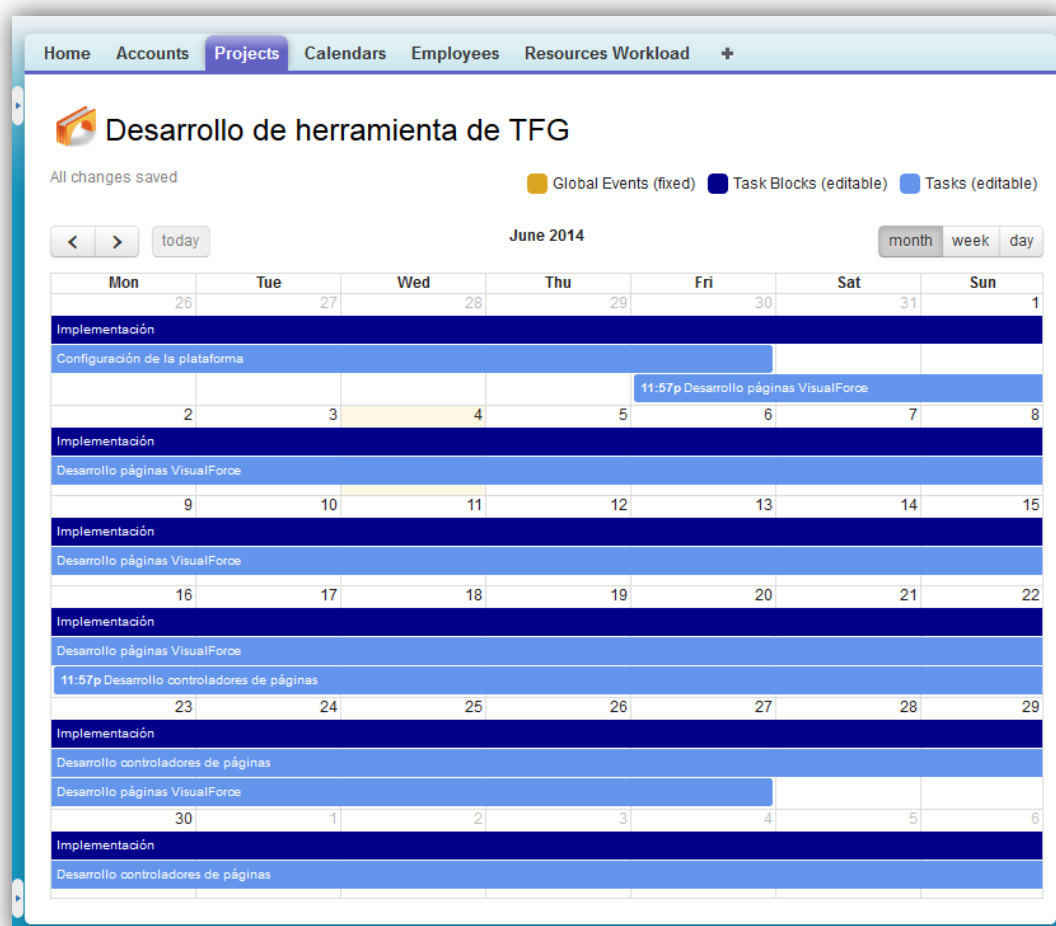


Figura 17. *Project: Vista de calendario*

Para añadir esta vista se ha utilizado FullCalendar [21], una librería jQuery de calendarios con muchas funcionalidades disponibles que ya había sido integrada en Salesforce por otro desarrollador [22]. Gracias a utilizar jQuery, se ha podido implementar en esta herramienta el arrastre de bloques y tareas que, mediante llamadas a métodos en el controlador Apex CalendarProjectExtension, se transmiten a los registros de la base de datos. Además, los nombres de los eventos representados son enlaces a las páginas de sus registros en el sistema.

3.3.1.4 CALENDARS

Esta es la página de visualización de calendarios, asignada a la cuarta pestaña. En ella se puede acceder a los registros de tipo calendario. Tras pasar la página de elementos recientes, se puede acceder a la vista del listado de calendarios.

Action	Calendar	Start Date	End Date	Is Default	Monday	Tuesday	Wednesday
<input type="checkbox"/> Edit Del	Calendar_1	1/01/2014	31/12/2014	<input type="checkbox"/>	480	480	480
<input type="checkbox"/> Edit Del	Calendar_2	1/01/2014	31/12/2014	<input checked="" type="checkbox"/>	8	8	8

Figura 18. Pestaña *Calendars*: listado

Al pulsar en uno de los registros, se accede a toda su información según el formato de página configurado. En este formato se muestra el valor de los campos de calendario divididos por secciones, así como las listas de excepciones y seguidores.

Calendar 1

Customize Page | Edit Layout | Printable View | Help for this Page

Back to List: Calendars

Exceptions (3) | Followers (6)

Calendar Detail [Edit] [Delete] [Clone] [Show Calendar]

▼ Information

Calendar	Calendar 1	End Date	31/12/2014
Start Date	1/01/2014	Is Default	<input type="checkbox"/>

▼ Week Days Working Times

Monday	480	Saturday	0
Tuesday	480	Sunday	0
Wednesday	480		
Thursday	480		
Friday	360		

▼ System Information

Created By	Borja Gonzalez Orduna, 19/02/2014 13:12	Last Modified By	Borja Gonzalez Orduna, 26/02/2014 10:30
Owner	Borja Gonzalez Orduna [Change]		

[Edit] [Delete] [Clone] [Show Calendar]

Exceptions [New Calendar Exception] [Exceptions Help]

Action	Calendar Exception	Start Date	End Date	Reason	Excepted Time	Full Day
[Edit] [Del]	Calendar Exception 1	31/01/2014 9:00	31/01/2014 12:30	Reson 1		<input checked="" type="checkbox"/>
[Edit] [Del]	Calendar Exception 2	18/03/2014 0:00	19/03/2014 23:59	Reson 2	470	<input type="checkbox"/>
[Edit] [Del]	Calendar Exception 3	1/05/2015 12:23	1/05/2015 12:23	May 2		<input checked="" type="checkbox"/>

Followers [Followers Help]

Action	Employee	Email
[Edit] [Del]	Borja González	borjagonord@gmail.com
[Edit] [Del]	Ackbar Calamarí	admiral@mail.moncal
[Edit] [Del]	Rafael Macías	rmacias@leadcllc.com
[Edit] [Del]	Juan José Soques	jisoques@leadcllc.com
[Edit] [Del]	Nora Pasero	npasero@leadcllc.com
[Edit] [Del]	Felipe Echanique	fechanique@leadcllc.com

Back To Top

Always show me fewer / more records per related list

Figura 19. Calendar: Formato de página

Se ha creado también un botón que abre una página personalizada con una ventana emergente de una vista de calendario en la que se muestran las excepciones asignadas.

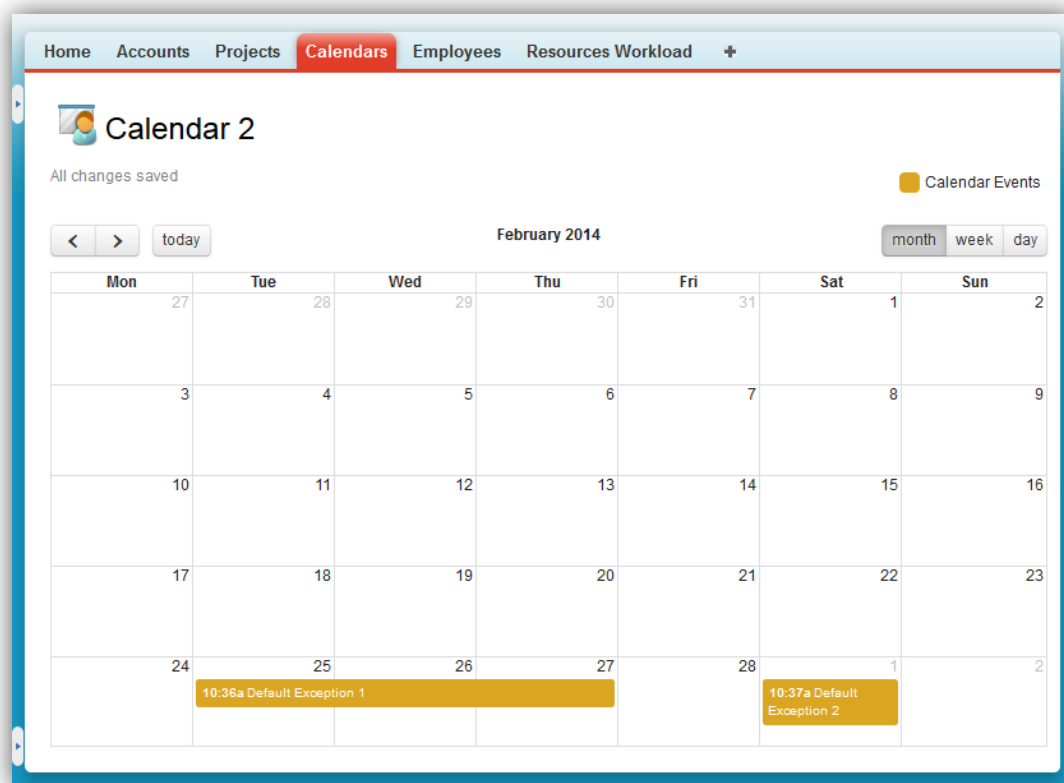


Figura 20. *Calendar*: Vista de calendario

Para añadir esta vista se ha utilizado FullCalendar [21], una librería jQuery. Se ha podido implementado el arrastre de excepciones de calendario que, mediante llamadas a métodos en el controlador Apex CalendarExtension, se transmiten a los registros de la base de datos. Además, los nombres de los eventos representados son enlaces a las páginas de sus registros en el sistema.

3.3.1.5 EMPLOYEES


Esta es la página de visualización de empleados, asignada a la quinta pestaña. En ella se puede acceder a los registros de tipo empleado. Tras pasar la página de elementos recientes, se puede acceder a la vista del listado de empleados.

Action	Employee	Email	Salesforce User
Edit Del	Ackbar Calamari	admiral@mail.moncal	
Edit Del	Borja González	borjaqonord@gmail.com	Borja Gonzalez Orduna
Edit Del	Felipe Echanique	fechanique@leadcllc.com	
Edit Del	Juan José Soques	jisoques@leadcllc.com	
Edit Del	Nora Pasero	npasero@leadcllc.com	
Edit Del	Rafael Macías	rmacias@leadcllc.com	

Figura 21. Pestaña *Employees*: listado

Al pulsar en uno de los registros, se accede a toda su información según el formato de página configurado. En este formato se muestra el valor de los campos de empleado divididos por secciones, así como las listas de excepciones de calendario personales, proyectos gestionados, tareas controladas, asignaciones de tareas y actividades de trabajo registradas.

Home
Accounts
Projects
Calendars
Employees
Resources Workload
+

Employee

Borja González

[Customize Page](#) | [Edit Layout](#) | [Printable View](#) | [Help for this Page](#)

[« Back to List: Custom Object Definitions](#)

[Personal Exceptions \(2\)](#) | [Managed Projects \(2\)](#) | [Handled Tasks \(5+\)](#) | [Task Assignments \(5+\)](#) | [Work Activities \(4\)](#)

Employee Detail
[Edit](#)
[Delete](#)
[Clone](#)
[Show Calendar](#)
[Send Email](#)

▼ Information

Employee **Borja González**
Salesforce User [Borja Gonzalez Orduna](#)

Labour Cost 20,00 €
Calendar [Calendar 1](#)

▼ Reporting Information

Email [borjaqonord@gmail.com](#)
Reporting Password *****

▼ System Information

Created By [Borja Gonzalez Orduna](#) 19/02/2014 17:24
Last Modified By [Borja Gonzalez Orduna](#) 4/06/2014 2:54

Owner [Borja Gonzalez Orduna](#) [\[Change\]](#)

[Edit](#)
[Delete](#)
[Clone](#)
[Show Calendar](#)
[Send Email](#)

Personal Exceptions
[New Calendar Exception](#)
[Personal Exceptions Help](#)

Action	Calendar Exception	Start Date	End Date	Reason	Excepted Time	Full Day
Edit Del	Personal Exception 1	19/02/2014 1:00	19/02/2014 1:00	Reason 1		<input checked="" type="checkbox"/>
Edit Del	Personal Exception 2	17/03/2014 0:00	17/03/2014 23:59	Reason 2	80	<input type="checkbox"/>

Managed Projects
[Managed Projects Help](#)

Action	Project	Account	Completion	Budget	Start Date	End Date	Is Closed
Edit Del	Desarrollo de herramienta de TFG	LeadClic Solutions	0	5.000,00 €	24/03/2014	10/11/2014	<input type="checkbox"/>
Edit Del	Test Project	LeadClic Solutions	60	25.000,00 €	12/03/2014	24/05/2014	<input type="checkbox"/>

Handled Tasks
[New Task](#)
[Handled Tasks Help](#)

Action	Task	Completion	Workload	Start Date	End Date
Edit Del	Corrección de errores	0	2.000	19/10/2014 0:58	8/11/2014 0:58
Edit Del	Implementación	0	2.000	13/09/2014 1:00	28/10/2014 1:00
Edit Del	Desarrollo Triggers	0	2.500	6/08/2014 23:57	11/09/2014 0:15
Edit Del	Desarrollo clases utilidades	0	1.500	28/07/2014 23:57	6/08/2014 23:57
Edit Del	Desarrollo de pruebas	0	1.000	25/07/2014 0:57	8/08/2014 0:29

[Show 5 more »](#) | [Go to list \(10\) »](#)

Task Assignments
[New Task Assignment](#)
[Task Assignments Help](#)

Action	Task Assignment Name	Task	Assigned Workload	Completion	Estimated Cost	Start Date	End Date
Edit Del	Desarrollo Trigger actualización de precedencias	Desarrollo Triggers	600	0	200,00 €	1/09/2014 0:24	12/09/2014 0:26
Edit Del	Desarrollo Trigger coste asignado	Desarrollo Triggers	400	0	133,33 €	21/08/2014 0:24	29/08/2014 0:24
Edit Del	Corrección	Corrección de errores	1.000	0	333,33 €	19/08/2014 11:12	7/09/2014 11:12
Edit Del	Desarrollo Trigger coste real	Desarrollo Triggers	200	0	66,67 €	14/08/2014 0:24	20/08/2014 0:24
Edit Del	Desarrollo Trigger fechas en tareas	Desarrollo Triggers	800	0	266,67 €	7/08/2014 0:24	14/08/2014 0:24

[Show 5 more »](#) | [Go to list \(16\) »](#)

Work Activities
[New Work Activity](#)
[Work Activities Help](#)

Action	Work Activity	Date	Time Spent	Task Assignment	Activity Cost
Edit Del	Test WA	20/05/2014	500	Test TA	166,67 €
Edit Del	WA-Task Assignment 1-2014-05-09 10:36:13Z	9/05/2014	150	Task Assignment 1	50,00 €
Edit Del	WA-Task Assignment 2-2014-04-23 07:54:31Z	23/04/2014	25	Task Assignment 2 Bo	8,33 €
Edit Del	WA1	16/04/2014	100	Task Assignment 1	33,33 €

[« Back To Top](#)

Always show me [▼ more](#) records per related list

Figura 22. Employee: Formato de página

Se han creado también dos botones personalizados, que abren sendas páginas personalizadas con una ventana emergente de una vista de calendario en la que se muestran las asignaciones de tareas y las excepciones personales y del calendario del empleado, y una interfaz de envío de correo electrónico.

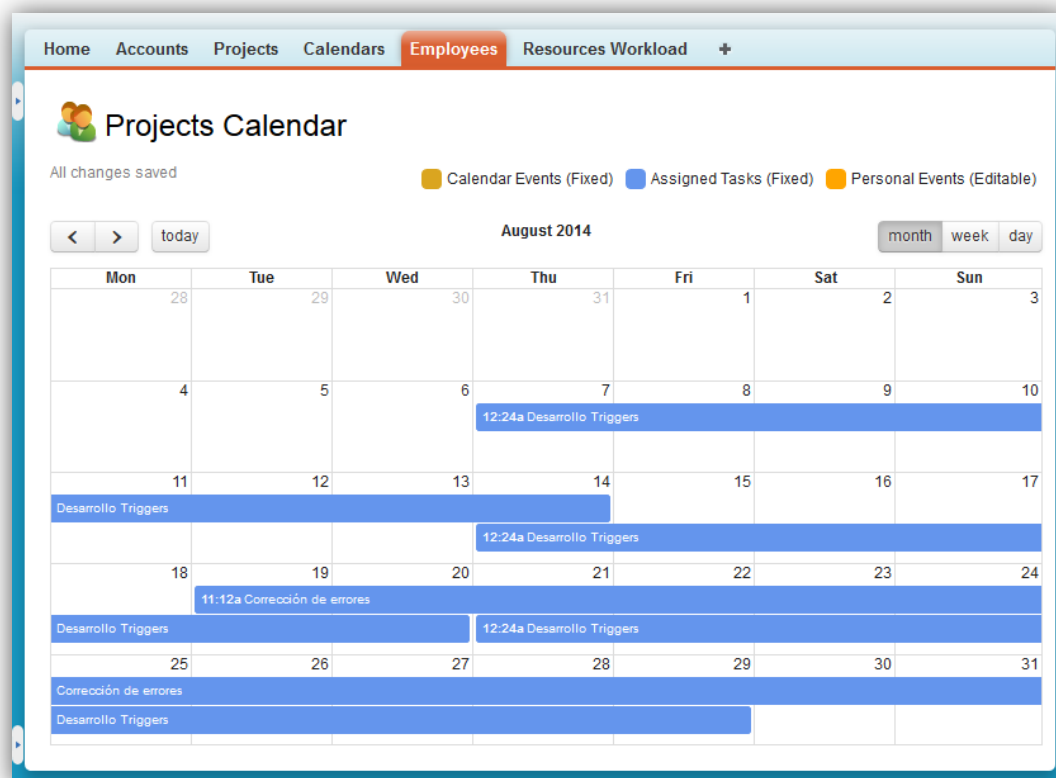
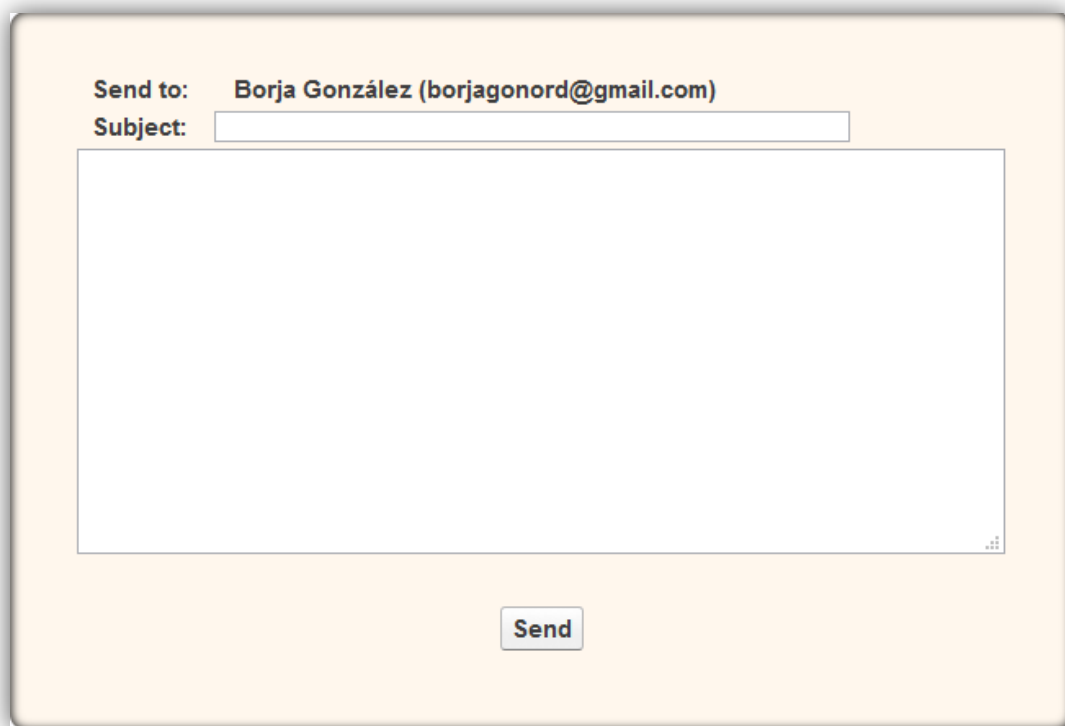


Figura 23. *Employee*: Vista de calendario

Para añadir la vista de calendario se ha utilizado FullCalendar [21], una librería jQuery. Se ha implementado en esta vista el arrastre de excepciones personales que, mediante llamadas a métodos en el controlador Apex CalendarEmployeeExtension, se transmiten a los registros de la base de datos. Además, los nombres de los eventos representados son enlaces a las páginas de sus registros en el sistema.



Send to: Borja González (borjagonord@gmail.com)

Subject:

Send

Figura 24. *Employee*: Envío de correo electrónico

La página de envío de correos electrónicos se ha programado haciendo uso del API de Salesforce para el envío de correos, realizando todo el procedimiento en el controlador Apex EmailExtension.

3.3.1.6 TASK BLOCKS

Esta es la página de visualización de bloques de tareas, que muestra toda la información de los registros de este tipo de objeto según el formato de página configurado. En este formato se muestra el valor de los campos de bloque de tareas divididos por secciones, así como las listas de tareas en que se divide y de relaciones de precedencia o dependencia con otros bloques y tareas.

The screenshot displays the 'Task Block' page for 'Análisis de requisitos'. The page is organized into several sections:

- Task Block Detail:** Includes buttons for 'Edit', 'Delete', and 'Clone'.
- Information:** A table showing task block details.

Task Block	Análisis de requisitos	Project	Desarrollo de herramienta de TFG
Start Date	30/03/2014 9:52		
End Date	18/04/2014 10:58	Tasks Workload	620
Assigned Cost	475,00 €	Assigned Workload	620
Incurred Cost	0,00 €	Completion	0
- System Information:** Shows 'Created By' as Boria Gonzalez Orduna (29/05/2014 9:29) and 'Last Modified By' as Boria Gonzalez Orduna (3/06/2014 0:04). Includes 'Edit', 'Delete', and 'Clone' buttons.
- Tasks:** A table listing tasks associated with the block.

Action	Task	Start Date	End Date	Workload
Edit Del	Toma de contacto	30/03/2014 9:52	4/04/2014 9:52	300
Edit Del	Análisis de requisitos	7/04/2014 10:56	11/04/2014 10:56	200
Edit Del	Negociación	14/04/2014 10:58	18/04/2014 10:58	120
- Tasks Precedences (Preceding Block):** Shows 'No records to display'.
- Tasks Precedences (Dependent Block):** Shows 'No records to display'.

At the bottom, there is a 'Back To Top' link and a note: 'Always show me fewer ▲ / ▼ more records per related list'.

Figura 25. Task Block: Formato de página

3.3.1.7 TASKS

Esta es la página de visualización de tareas, que muestra toda la información de los registros de este tipo de objeto según el formato de página configurado. En este formato se muestra el valor de los campos de tarea divididos por secciones, así como las listas de asignaciones a empleados y de relaciones de precedencia o dependencia con otros bloques y tareas.

The screenshot displays the 'Task' page for 'Análisis de requisitos'. The page is divided into several sections:

- Task Detail:** Includes buttons for 'Edit', 'Delete', 'Clone', and 'Recalculate Costs'.
- Information:** A table showing task details:

Task	Análisis de requisitos	Task Block	Análisis de requisitos
Handler	Juan José Soques	Related Objective	Definición precisa de requisitos
Start Date	7/04/2014 10:56	Base Start Date	
End Date	11/04/2014 10:56	Base End Date	
Workload	200	Assigned Cost	125,00 €
Assigned Workload	200	Incurring Cost	0,00 €
Completion	0		
Is Completed	<input type="checkbox"/>		
- System Information:** Shows 'Created By' as Boria Gonzalez Orduna (29/05/2014 10:58) and 'Last Modified By' as Boria Gonzalez Orduna (3/06/2014 0:01). It also includes 'Edit', 'Delete', 'Clone', and 'Recalculate Costs' buttons.
- Task Assignments:** A table showing assignments:

Action	Task Assignment Name	Assigned Workload	Completion	Employee	Start Date	End Date	Base Start Date	Base End Date
Edit Del	Análisis de requisitos	100	0	Juan José Soques	7/10/2013 10:56	11/10/2013 10:56		
Edit Del	Análisis de requisitos	100	0	Nora Pasero	7/10/2013 10:56	11/10/2013 10:56		
- Tasks Precedences (Preceding Task):** A section with a 'New Tasks Precedence' button and a message 'No records to display'.
- Tasks Precedences (Dependent Task):** A section with a 'New Tasks Precedence' button and a message 'No records to display'.

At the bottom, there is a 'Back To Top' link and a note: 'Always show me fewer / more records per related list'.

Figura 26. Task: Formato de página

Se ha creado un botón que recalcula los costes de las actividades de trabajo relacionadas, lo que actualiza también los costes reales de asignaciones, tarea, bloque y proyecto, para el caso en que se modifique el salario de un empleado con asignaciones de esta tarea.

3.3.1.8 TASK PRECEDENCES

Esta es la página de visualización de precedencias de tarea, que muestra toda la información de los registros de este tipo de objeto según el formato de página configurado. En este formato se muestra el valor de los campos de precedencia de tareas divididos por secciones.

Home Accounts Projects Calendars Employees Resources Workload +

Tasks Precedence [Edit Layout](#) | [Printable View](#) | [Help for this Page](#)

Análisis sobre diseño

[Back to List: Pages](#)

Tasks Precedence Detail [Edit](#) [Delete](#) [Clone](#)

▼ Information

Tasks Precedence	Análisis sobre diseño	Offset Minutes
Preceding Task		Dependent Task
Preceding Block	Análisis de requisitos	Dependent Block
		Diseño de la herramienta
Precedent End Date	18/04/2014 10:58	Dependent Start Date
		18/04/2014 10:58

▼ System Information

Created By	Boria Gonzalez Orduna	4/06/2014 19:12	Last Modified By	Boria Gonzalez Orduna	4/06/2014 19:12
Owner	Boria Gonzalez Orduna Change				

[Edit](#) [Delete](#) [Clone](#)

Figura 27. Task Precedence: Formato de página

3.3.1.9 TASK ASSIGNMENTS

Esta es la página de visualización de asignaciones de tareas, que muestra toda la información de los registros de este tipo de objeto según el formato de página configurado. En este formato se muestra el valor de los campos de asignaciones de tareas divididos por secciones.

The screenshot displays a web application interface for task assignments. At the top, there is a navigation bar with links: Home, Accounts, Projects, Calendars, Employees, Resources, Workload, and a plus icon. Below the navigation bar, the page title is 'Task Assignment' with a star icon, followed by 'Diseño del modelo de datos'. There are links for 'Customize Page', 'Edit Layout', 'Printable View', and 'Help for this Page'. A 'Back to List: Pages' link is also present. Below the title, there is a 'Work Activities [1]' link. The main section is titled 'Task Assignment Detail' and includes buttons for 'Edit', 'Delete', 'Clone', and 'Calculate End Date'. The 'Information' section shows details for the task assignment, including the task name, employee, assigned workload, completion status, start and end dates, and estimated and real costs. The 'System Information' section shows the creation and last modification details. At the bottom, there is a 'Work Activities' table with columns for Action, Work Activity, Date, Time Spent, and Activity Cost. The table contains one row for the task 'Diseño del modelo de datos' with a date of 18/04/2014, 60 time spent, and a cost of 20,00 €. There are also links for 'New Work Activity' and 'Work Activities Help'.

Action	Work Activity	Date	Time Spent	Activity Cost
Edit Del	Diseño del modelo de datos	18/04/2014	60	20,00 €

Figura 28. Task Assignment: Formato de página

Se ha creado un botón que recalcula la fecha prevista de finalización de la asignación basándose en la carga de trabajo del empleado y de la asignación.

3.3.1.10 CALENDAR EXCEPTIONS

Esta es la página de visualización de excepciones de calendario, que muestra toda la información de los registros de este tipo de objeto según el formato de página configurado. En este formato se muestra el valor de los campos de excepciones de tareas divididos por secciones.

The screenshot displays a web application interface for managing calendar exceptions. The top navigation bar includes links for Home, Accounts, Projects, Calendars, Employees, Resources, and Workload. The main header shows the title 'Calendar Exception' and a sub-header 'Holiday'. Below the header, there are buttons for 'Edit', 'Delete', and 'Clone'. The page is divided into two main sections: 'Information' and 'System Information'. The 'Information' section contains fields for 'Calendar Exception' (Holiday), 'Start Date' (15/11/2013 18:28), 'End Date' (15/11/2013 18:28), 'Reason' (Company's holiday), 'Excepted Time', 'Full Day' (checked), 'Employee', and 'Calendar' (Calendar 2). The 'System Information' section shows 'Created By' (Boria Gonzalez Orduna, 2/06/2014 18:29), 'Last Modified By' (Boria Gonzalez Orduna, 5/06/2014 1:25), and 'Owner' (Boria Gonzalez Orduna [Change]).

Calendar Exception Detail	
Information	
Calendar Exception	Holiday
Start Date	15/11/2013 18:28
End Date	15/11/2013 18:28
Reason	Company's holiday
Excepted Time	
Full Day	<input checked="" type="checkbox"/>
Employee	
Calendar	Calendar 2
System Information	
Created By	Boria Gonzalez Orduna, 2/06/2014 18:29
Last Modified By	Boria Gonzalez Orduna, 5/06/2014 1:25
Owner	Boria Gonzalez Orduna [Change]

Figura 29. Calendar Exception: Formato de página

3.3.1.11 WORK ACTIVITIES

Esta es la página de visualización de actividades de trabajo, que muestra toda la información de los registros de este tipo de objeto según el formato de página configurado. En este formato se muestra el valor de los campos de actividades de trabajo divididos por secciones.

The screenshot displays a web application interface for 'Work Activity'. The top navigation bar includes links for Home, Accounts, Projects, Calendars, Employees, Resources, and Workload. The main header shows the title 'Work Activity' and the specific record 'WA-Página de análisis de proyecto-2014-06-04 18:29:18Z'. Below the header, there are buttons for 'Edit', 'Delete', and 'Clone'. The page is divided into two main sections: 'Information' and 'System Information'. The 'Information' section contains fields for 'Work Activity' (WA-Página de análisis de proyecto-2014-06-04 18:29:18Z), 'Task Assignment' (Página de análisis de proyecto), 'Employee' (Boria González), 'Date' (25/06/2014), 'Time Spent' (150), and 'Activity Cost' (50,00 €). The 'System Information' section contains fields for 'Created By' (Boria Gonzalez Orduna, 4/06/2014 20:29), 'Last Modified By' (Boria Gonzalez Orduna, 4/06/2014 20:29), and 'Owner' (Boria Gonzalez Orduna [Change]). At the bottom of the 'System Information' section, there are buttons for 'Edit', 'Delete', and 'Clone'.

Work Activity Detail	
Information	
Work Activity	WA-Página de análisis de proyecto-2014-06-04 18:29:18Z
Task Assignment	Página de análisis de proyecto
Employee	Boria González
Date	25/06/2014
Time Spent	150
Activity Cost	50,00 €
System Information	
Created By	Boria Gonzalez Orduna , 4/06/2014 20:29
Last Modified By	Boria Gonzalez Orduna , 4/06/2014 20:29
Owner	Boria Gonzalez Orduna [Change]

Figura 30. Work Activity: Formato de página

3.3.2 ANÁLISIS Y GESTIÓN DE PROYECTOS

La página de análisis de proyecto, desarrollada totalmente mediante el lenguaje Visualforce con controladores Apex, es la más compleja e importante de la herramienta y debe facilitar la labor de los gestores de proyectos de la empresa.

Para un proyecto dado, muestra una cabecera de información básica y una serie de secciones con información dividida por temas. Se ha desarrollado en una sola página, modificando la visibilidad de los componentes, de manera que sólo deba cargarse una vez la interfaz y después las transiciones entre secciones sean más rápidas.

Para llegar a la página de análisis de un proyecto, se puede pulsar sobre el botón que aparece en la página de los registros de proyecto.

Esta página utiliza varios controladores Apex para manipular y organizar la información:

- **ProjectAnalysisExtension:** obtiene y organiza la información de la base de datos referente a empleados y tareas, y efectúa la mayoría operaciones de la interfaz, como creación y eliminación de registros, control de visibilidad, generación de parámetros de algunas gráficas o envío de correos electrónicos.
- **CalendarProjectExtension:** obtiene y organiza la información de la base de datos relativa a tareas, bloques y excepciones de calendario para generar los parámetros del calendario del proyecto, y efectúa las operaciones de desplazamientos de tareas y bloques realizados desde el calendario.
- **ResourcesWorkloadController:** obtiene la información de la base de datos relacionada con empleados, calendarios de trabajo, asignaciones de tareas y excepciones para generar los parámetros de construcción de las gráficas de cargas de trabajo en la pestaña de recursos involucrados.

3.3.2.1 CABECERA DE PROYECTO

La cabecera es una sección estática que se muestra siempre en la parte superior de la página. En ella se muestran los datos más relevantes del proyecto a gestionar, así como los botones de navegación entre secciones de la página.

The screenshot shows a web application interface with a navigation bar at the top containing links: Home, Accounts, Projects (highlighted), Calendars, Employees, Resources, Workload, and a plus icon. Below the navigation bar, the main content area displays the project header for 'Desarrollo de herramienta de TFG' (LeadClic Solutions). The header is organized into a grid of fields:

Manager ⓘ Boria González	Budget ⓘ 5.000,00 €	Assigned Cost ⓘ 5.031,24 €	Incurring Cost ⓘ 566,67 €
Start Date ⓘ 24/03/2014	End Date ⓘ 10/11/2014	Workload 350 h	Completion 4 %
Tasks Start Date ⓘ 30/03/2014 9:52	Tasks End Date ⓘ 8/11/2014 0:58	Is Closed ⓘ <input type="checkbox"/>	

At the bottom of the header section, there are five buttons for navigation: 'Completion / EVM', 'Gantt Diagram', 'Calendar View', 'Tasks List', and 'Involved Resources'.

Figura 31. Análisis de proyectos: Cabecera

Algunos de los valores que aparecen aquí son extraídos directamente del registro de proyecto, mientras que otros son calculados en tiempo de ejecución por el controlador.

3.3.2.2 ESTADO DE COMPLETITUD Y EVM

Esta pestaña está dividida en dos secciones, una para el estado de completitud y otra para los valores de la metodología de Gestión del Valor Ganado (EVM) [10].

En la primera sección se muestra la completitud del proyecto en cuanto a tiempo de trabajo, calculado según las cargas de trabajo asignadas a las tareas y la completitud de cada una, según se ha informado por parte de los empleados. Aparecen los datos numéricamente y en forma de gráfica que representa el progreso del proyecto a lo largo del tiempo, según la carga de trabajo programada, asignada a empleados y completada, todas medidas en minutos.

Para el cómputo de carga de trabajo completada sólo se utilizan las cargas de trabajo de las tareas terminadas.

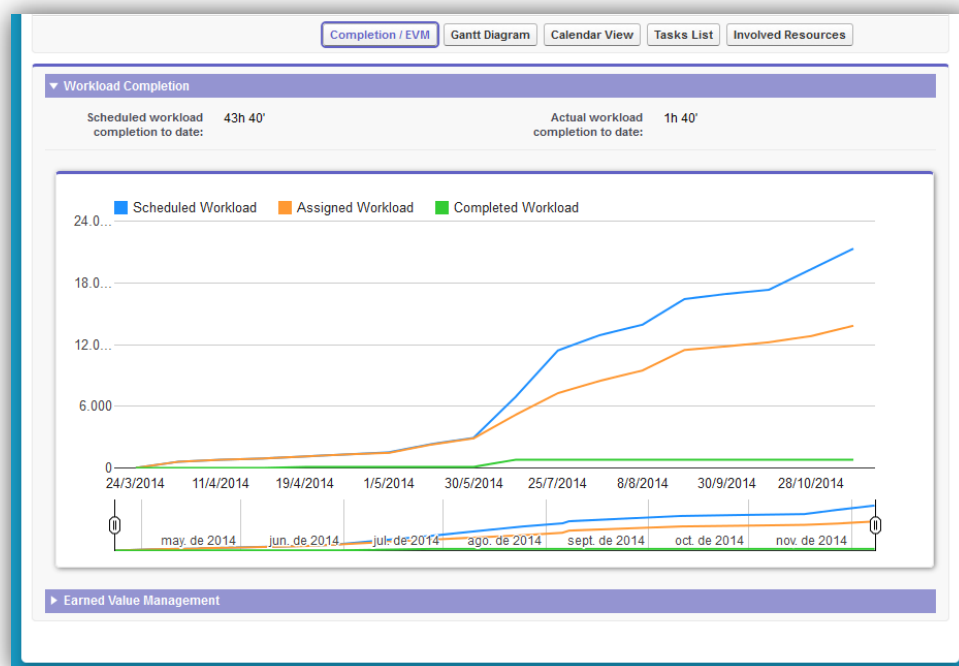


Figura 32. Análisis de proyectos: Completitud

En la segunda sección se muestra el análisis EVM del proyecto. Se ha escogido esta metodología ya que ofrece indicadores muy potentes sobre el estado de un proyecto en tiempo real, así como predicciones fiables a futuro basadas en los datos disponibles. La metodología EVM sólo tiene en cuenta tareas completadas.

En primer lugar aparecen listados los parámetros más importantes de la metodología, calculados al momento, y a continuación la gráfica que muestra el avance de los parámetros BCWS, BCWP y ACWP³ a lo largo del tiempo.

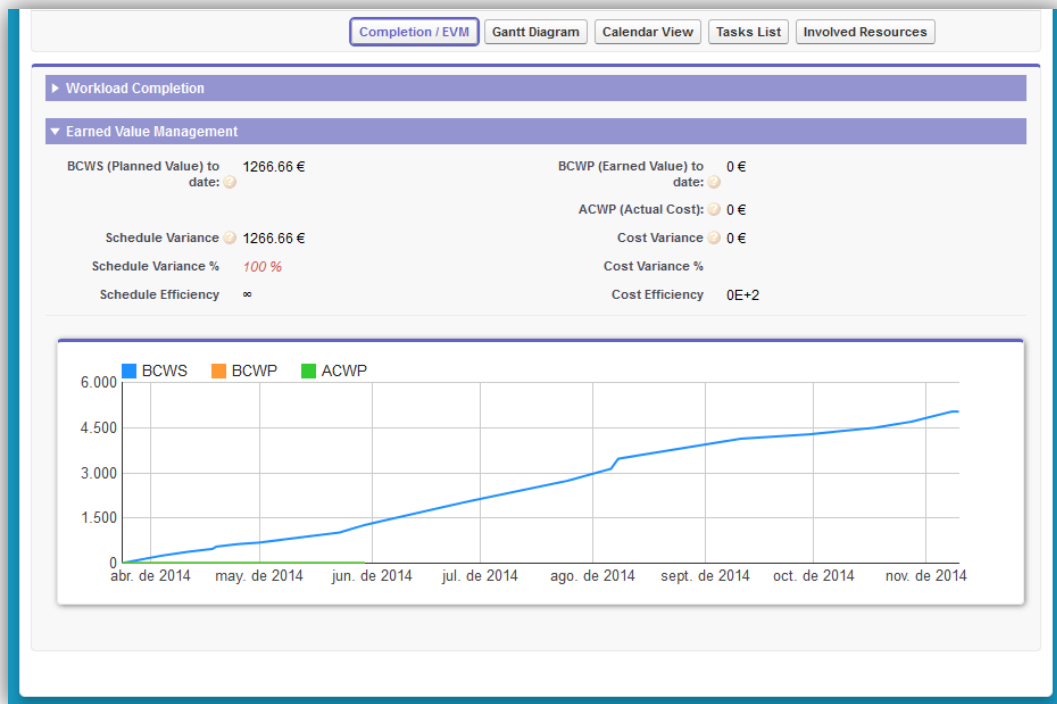


Figura 33. Análisis de proyectos: Gestión del Valor Ganado (EVM)

Para mostrar estas gráficas se ha utilizado la librería de gráficas de Google para desarrolladores [23], a la cual se puede acceder externamente mediante Javascript, construyendo los parámetros necesarios en el código del controlador.

³ BCWS: *Budgeted Cost for Work Scheduled*, valor presupuestado para el trabajo previsto hasta la fecha.
 BCWP: *Budgeted Cost for Work Performed*, valor presupuestado para el trabajo realizado hasta la fecha.
 ACWP: *Actual Cost of Work Performed*: coste real del trabajo realizado hasta la fecha.

3.3.2.3 DIAGRAMA DE GANTT

El diagrama de Gantt es una gráfica que representa el avance de las tareas de un proyecto respecto al tiempo. En el diagrama aparecen dos paneles verticales: el panel a la izquierda muestra la lista de tareas del proyecto, en orden cronológico de arriba abajo y agrupadas según bloques, y el panel a la derecha muestra su duración en el tiempo mediante barras horizontales. Cada nombre de tarea o bloque es un enlace al registro en la base de datos, para acceder al acceso a la información, y al pasar el ratón por encima de las barra se muestra un breve resumen con la información más importante.

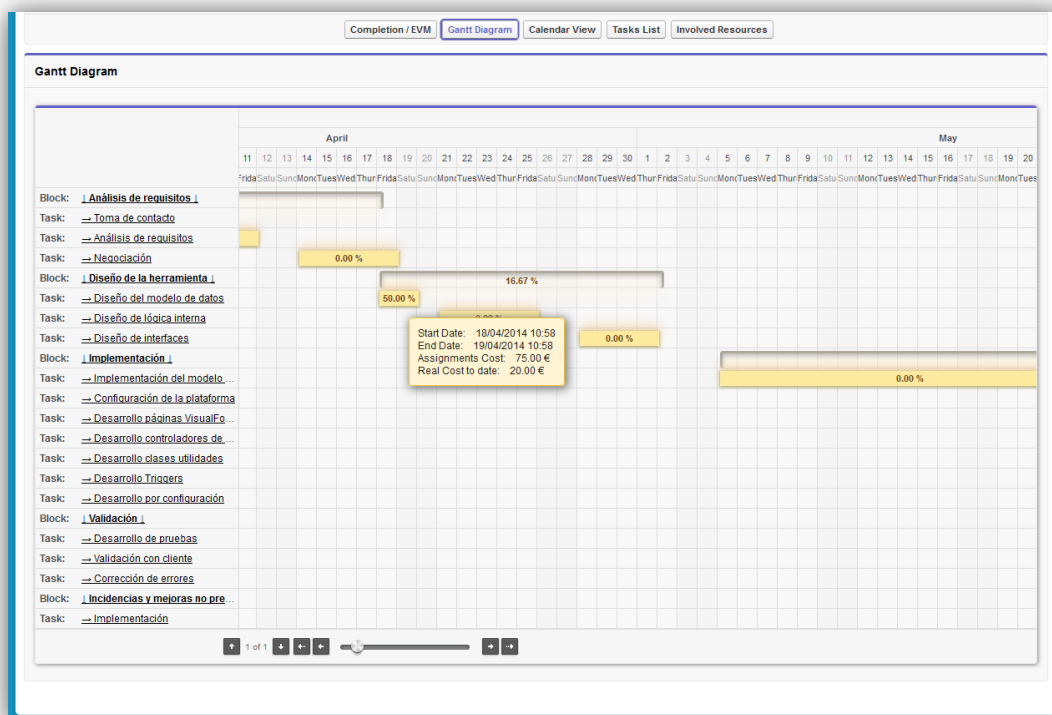


Figura 34. Análisis de proyectos: Diagrama de Gantt

Para el desarrollo de esta vista se ha partido de una librería externa Javascript libre [24], parcialmente integrada una versión antigua en Salesforce por otro desarrollador [25], que ha sido ampliamente adaptada y modificada para mejorar sus características y satisfacer las necesidades de esta herramienta.

3.3.2.4 VISTA DE CALENDARIO

En esta sección se muestra un calendario del proyecto, en el que se representan bloques de tareas, tareas y excepciones del calendario del sistema declarado como estándar.

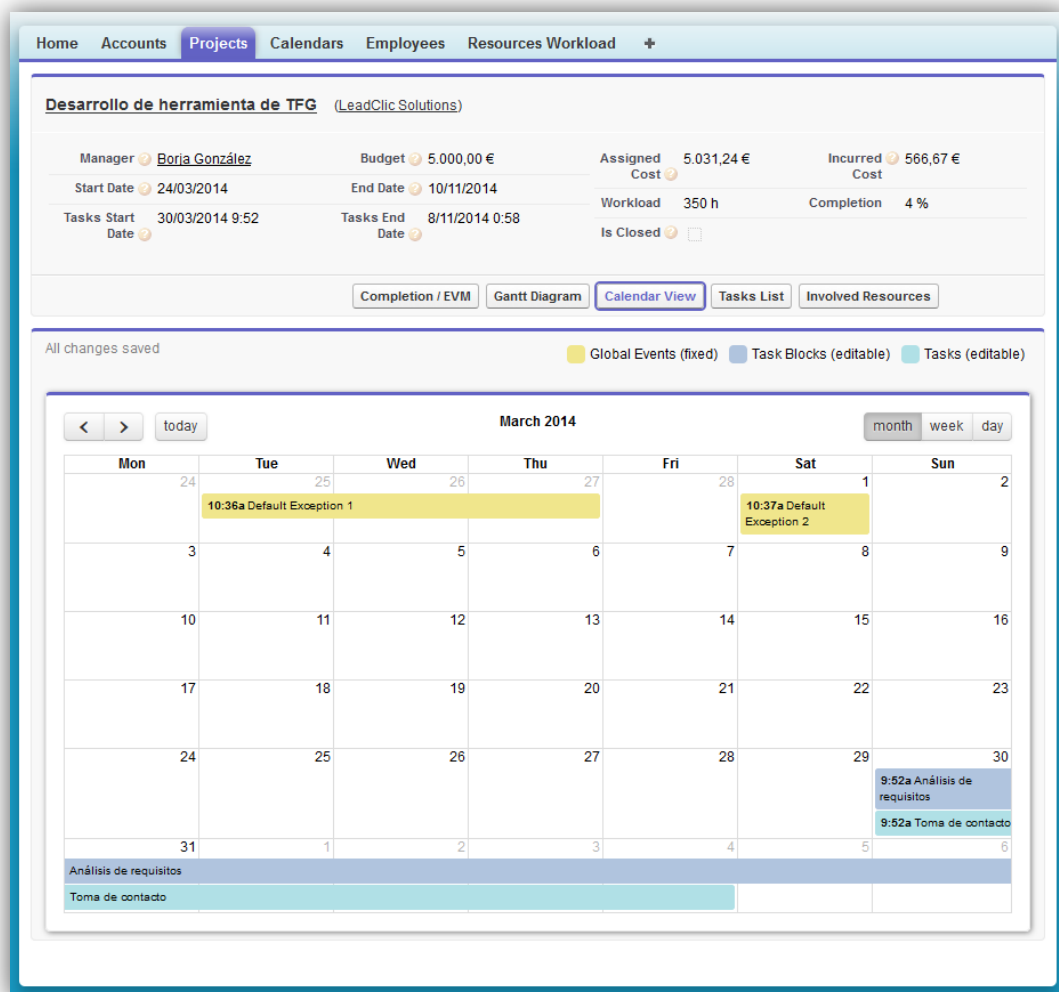


Figura 35. Análisis de proyectos: Vista de calendario

Para añadir esta vista se ha utilizado FullCalendar [21], una librería jQuery. Se ha implementado en el arrastre de tareas y bloques que, mediante llamadas a métodos en

el controlador Apex, se transmiten a los registros de la base de datos manteniendo las relaciones de precedencia y pertenencia a bloques.

Así, por ejemplo, si se arrastrara un bloque de tareas a otra fecha, todas las tareas pertenecientes al mismo, así como las tareas y bloques que dependieran de él, se desplazarían de la misma manera, propagándose los cambios y actualizándose el calendario instantes después. Además, el estado de la transacción se ha representado en un mensaje para que el usuario conozca el resultado de la operación y el momento en que finaliza. Se muestra un ejemplo en las figuras siguientes.

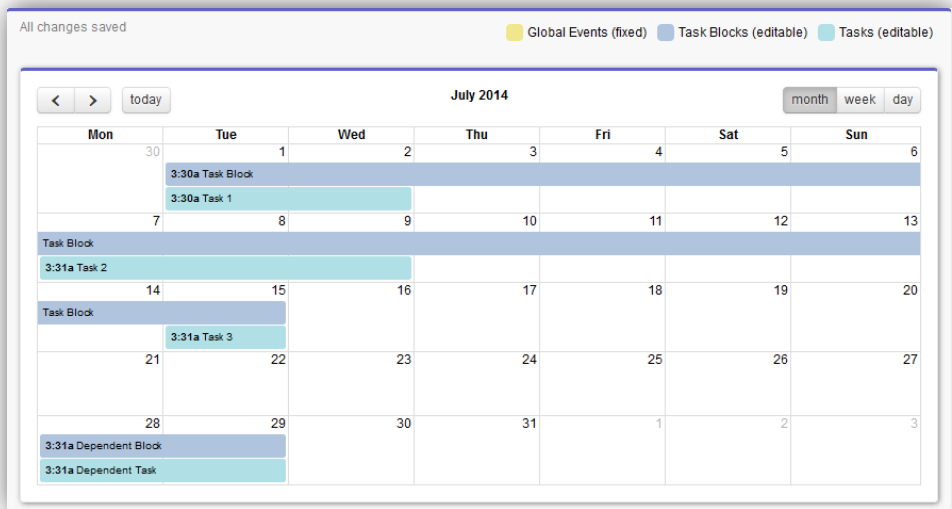


Figura 36. Estado inicial de calendario

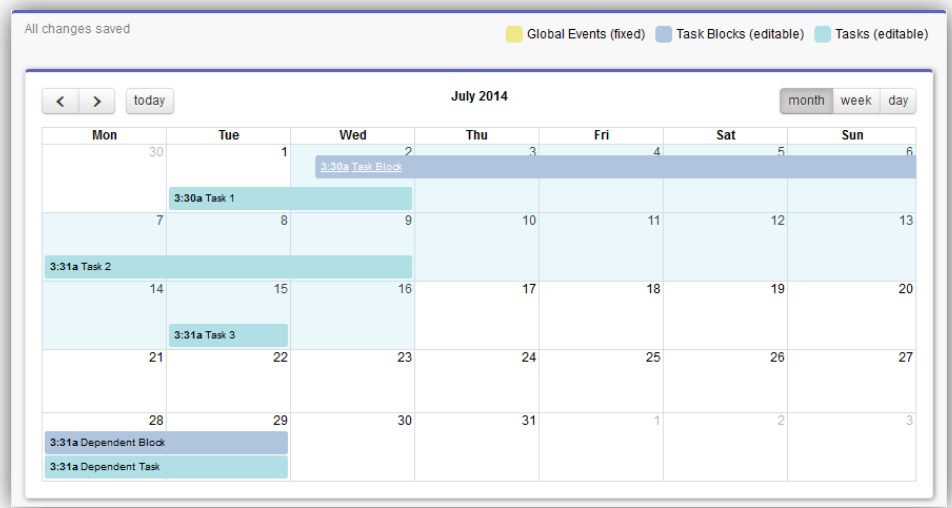


Figura 37. Se arrastra un bloque a un día posterior

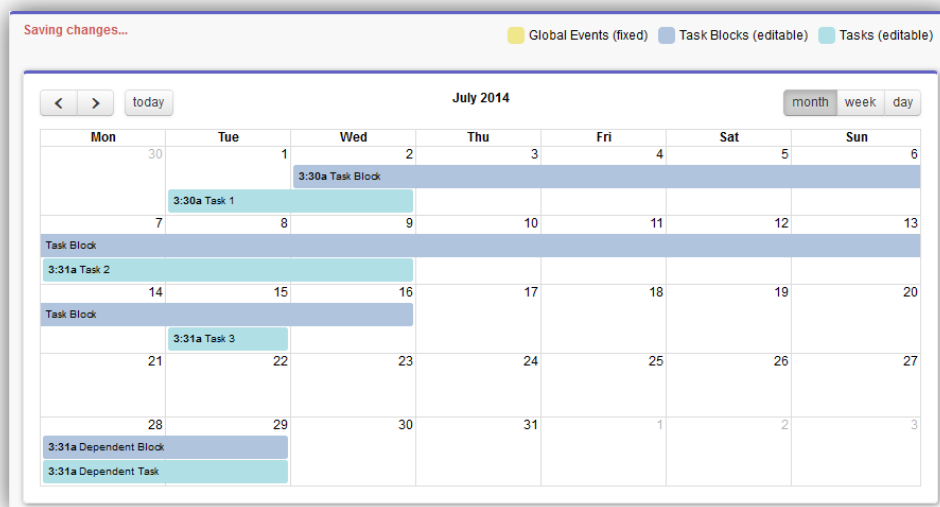


Figura 38. Al soltar el bloque, comienzan a procesarse los cambios en el controlador

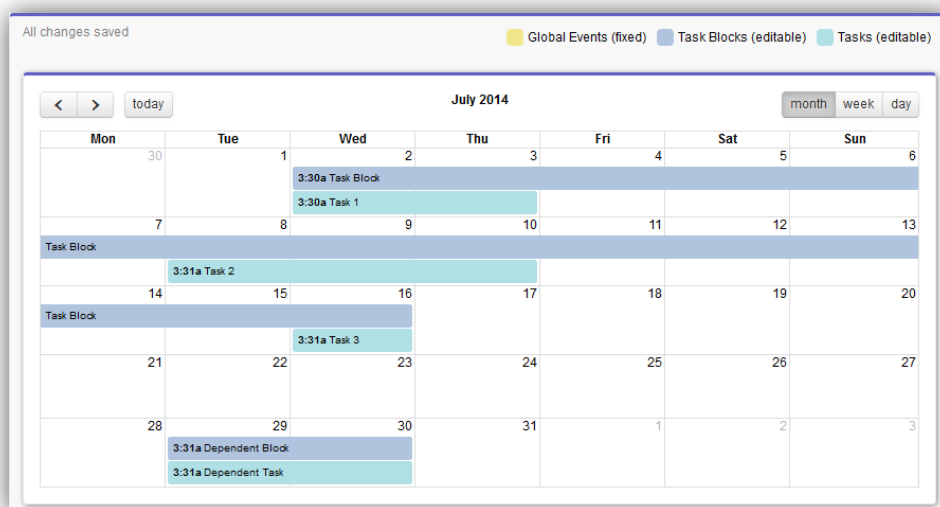


Figura 39. Al finalizar, las tareas del bloque y la tarea del bloque dependiente se han desplazado un día

Por otra parte, todos los nombres de tareas y bloques son enlaces a sus correspondientes registros en el sistema para facilitar el acceso a la información.

3.3.2.5 LISTADO DE TAREAS

Para facilitar una vista más básica del proyecto, en esta pestaña se muestra un listado de tareas, agrupadas en tablas por bloques. Se muestra para cada tarea la información esencial así como la variación sobre el coste previsto para la misma. Todas las referencias a bloques, tareas y empleados son enlaces a sus registros correspondientes en la base de datos para facilitar el acceso.







Completion / EVM Gantt Diagram Calendar View Tasks List Involved Resources									
✚ Add Task Block									
▼ Análisis de requisitos (10h 20') - From 30/03/2014 9:52 to 18/04/2014 10:58 Assigned/Real Costs: 475,00 €/0,00 € Delete Task Block									
Task Name	Start Date	End Date	Workload	Assigned Workload	Assignments Cost	Real Cost to Date	Completion	Handler	Action
Toma de contacto	30/03/2014 9:52	4/04/2014 9:52	5h 0'	5h 0'	250,00 €	0,00 € (-250,00 €)	0 %	Juan José Soques	
Análisis de requisitos	7/04/2014 10:56	11/04/2014 10:56	3h 20'	3h 20'	125,00 €	0,00 € (-125,00 €)	0 %	Juan José Soques	
Negociación	14/04/2014 10:58	18/04/2014 10:58	2h 0'	2h 0'	100,00 €	0,00 € (-100,00 €)	0 %	Juan José Soques	
✚ Add Task									
▼ Diseño de la herramienta (10h 0') - From 18/04/2014 10:58 to 1/05/2014 23:57 Assigned/Real Costs: 208,33 €/20,00 € Delete Task Block									
Task Name	Start Date	End Date	Workload	Assigned Workload	Assignments Cost	Real Cost to Date	Completion	Handler	Action
Diseño del modelo de datos	18/04/2014 10:58	19/04/2014 10:58	3h 20'	3h 20'	75,00 €	20,00 € (-55,00 €)	50 %	Nora Pasero	
Diseño de lógica interna	21/04/2014 23:57	25/04/2014 23:57	3h 20'	3h 20'	83,33 €	0,00 € (-83,33 €)	0 %	Nora Pasero	
Diseño de interfaces	28/04/2014 23:57	1/05/2014 23:57	3h 20'	2h 30' (-0h 50')	50,00 €	0,00 € (-50,00 €)	0 %	Nora Pasero	
✚ Add Task									
▶ Implementación (240h 0') - From 5/05/2014 23:57 to 30/09/2014 23:57 Assigned/Real Costs: 3.262,50 €/546,67 € Delete Task Block									
▶ Validación (56h 40') - From 25/07/2014 0:57 to 8/11/2014 0:58 Assigned/Real Costs: 885,41 €/0,00 € Delete Task Block									
▶ Incidencias y mejoras no previstas (33h 20') - From 13/09/2014 1:00 to 28/10/2014 1:00 Assigned/Real Costs: 200,00 €/0,00 € Delete Task Block									

Figura 40. Análisis de proyectos: Lista de tareas

Se han incluido además botones y enlaces para añadir y eliminar bloques y tareas, así como una funcionalidad de envío de correos electrónicos a los responsables

de las tareas para facilitar la comunicación en el equipo. Al pulsar sobre el icono del sobre, se abre una interfaz de envío de correos a la dirección almacenada para el usuario, evitando en todo momento abandonar la página desde la que se ha decidido la acción.

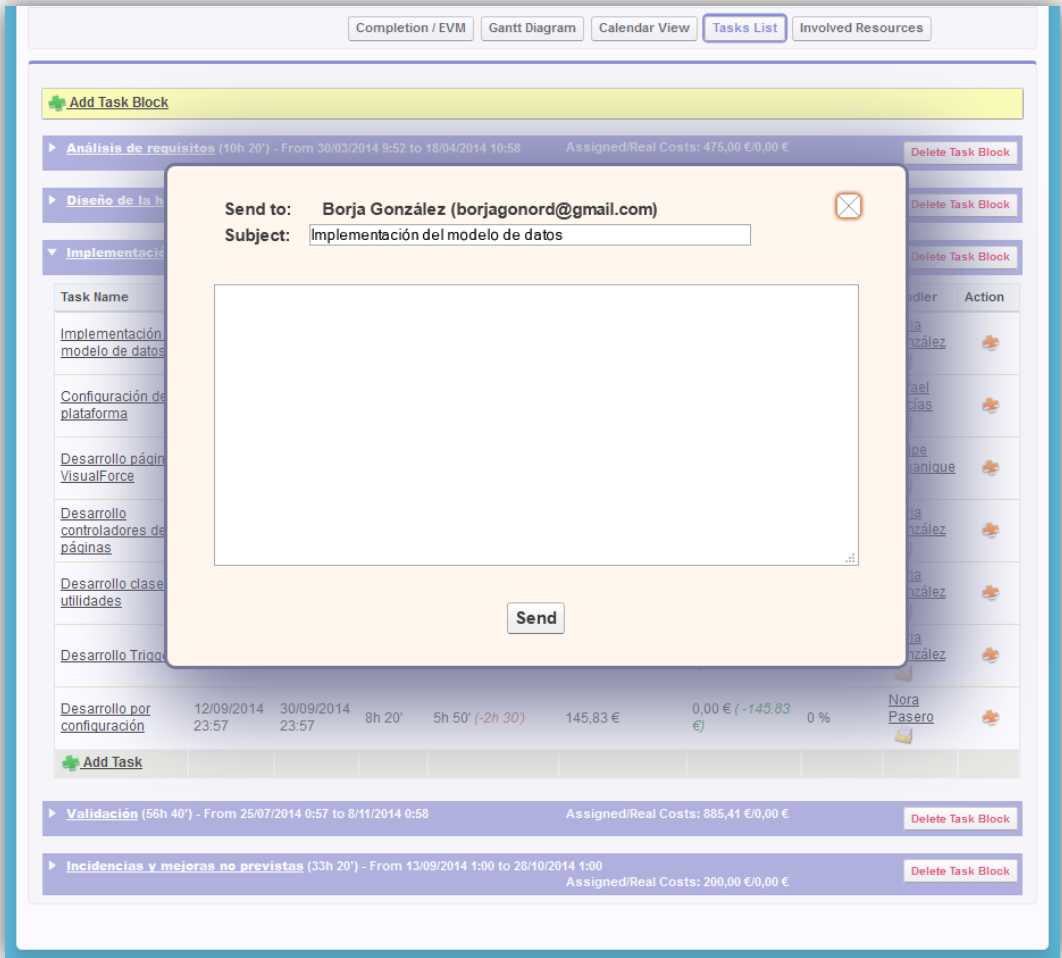


Figura 41. Envío de correo sin abandonar interfaz

También es posible, al situar el ratón sobre el nombre de un empleado, acceder a un enlace tipo *mailto* que abrirá la aplicación de mensajería por defecto del usuario para enviar el correo desde la misma.

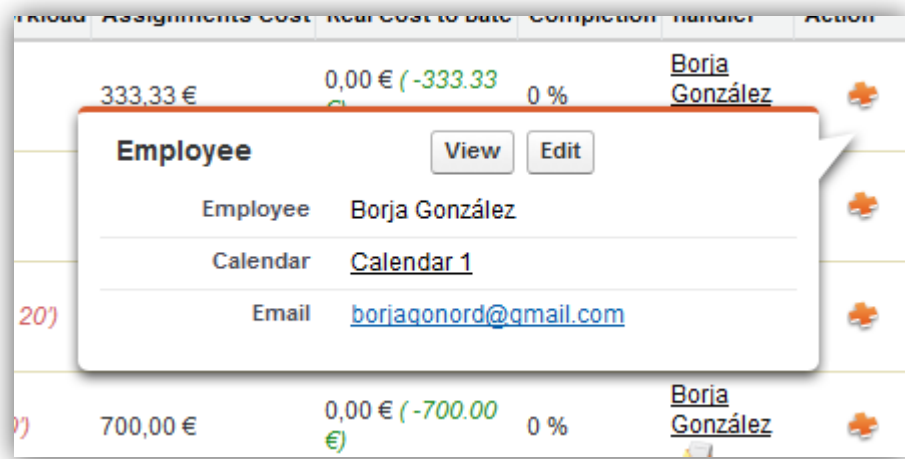


Figura 42. Envío de correos con aplicación de mensajería

3.3.2.6 RECURSOS HUMANOS INVOLUCRADOS

En la última pestaña se puede consultar un listado de los empleados que participan en el proyecto, es decir, aquellos que tienen tareas asignadas. Todas las referencias a registros del sistema son enlaces a los mismos para facilitar el acceso. Para cada empleado se muestra una lista de asignaciones de tareas de este proyecto, con la información esencial, desde la que se pueden añadir o eliminar más asignaciones. Además, al igual que en la sección anterior (Figura 41. Envío de correo sin abandonar interfaz), se pueden enviar correos electrónicos a los empleados al pulsar sobre el icono del sobre.

Completion / EVM Gantt Diagram Calendar View Tasks List Involved Resources						
▼ Borja González [Contact user] 📧						
Task Assignments	Related Task	Assigned Workload	Start Date	End Date	Completion	Action
Diseño del modelo de datos	Diseño del modelo de datos	1h 40'	18/04/2014 10:58	19/04/2014 10:58	100%	📧
Desarrollo controlador reporting de empleados	Desarrollo controladores de páginas	13h 20'	16/06/2014 0:21	27/06/2014 0:21	0%	📧
Página de análisis de proyecto	Desarrollo páginas VisualForce	16h 40'	23/06/2014 11:02	27/06/2014 0:18	700%	📧
Desarrollo controlador carga de trabajo	Desarrollo controladores de páginas	6h 40'	30/06/2014 0:21	4/07/2014 0:21	0%	📧
Desarrollo controlador calendario	Desarrollo controladores de páginas	6h 40'	7/07/2014 0:21	11/07/2014 0:21	0%	📧
Desarrollo controlador análisis proyectos	Desarrollo controladores de páginas	8h 20'	14/07/2014 0:21	25/07/2014 0:21	0%	📧
Movimiento de tareas	Desarrollo clases utilidades	16h 40'	2/08/2014 0:23	6/08/2014 0:23	0%	📧
Desarrollo Trigger fechas en tareas	Desarrollo Triggers	13h 20'	7/08/2014 0:24	14/08/2014 0:24	0%	📧
Desarrollo Trigger coste real	Desarrollo Triggers	3h 20'	14/08/2014 0:24	20/08/2014 0:24	0%	📧
Corrección	Corrección de errores	16h 40'	19/08/2014 11:12	7/09/2014 11:12	0%	📧
Desarrollo Trigger coste asignado	Desarrollo Triggers	6h 40'	21/08/2014 0:24	29/08/2014 0:24	0%	📧
Desarrollo Trigger actualización de precedencias	Desarrollo Triggers	10h 0'	1/09/2014 0:24	12/09/2014 0:26	0%	📧
Gestión de incidencias	Implementación	10h 0'	13/09/2014 11:14	28/10/2014 11:14	0%	📧
➕ Add assignment						
📅 Show/Hide Workload Calendar						
▶ Felipe Echanique [Contact user] 📧						
▶ Juan José Soques [Contact user] 📧						
▶ Nora Pasero [Contact user] 📧						
▶ Rafael Macías [Contact user] 📧						

Figura 43. Análisis de proyectos: Recursos involucrados

Para facilitar una decisión informada en la asignación de tareas a empleados, se ha incluido una gráfica desplegable en la que se muestra sobre un calendario la carga de trabajo disponible de cada empleado. Se representa mediante un código de color la disponibilidad para cada fecha (azul más intenso, mayor disponibilidad; rojo más intenso, menor disponibilidad). Además, se puede conocer la carga exacta (medida en minutos) en cada fecha al poner el cursor sobre la casilla. Para realizar este cálculo se tienen en cuenta las asignaciones del empleado en todos los proyectos de la empresa, lo que es adecuado para un equipo que se movilice entre varios proyectos según las necesidades del momento.

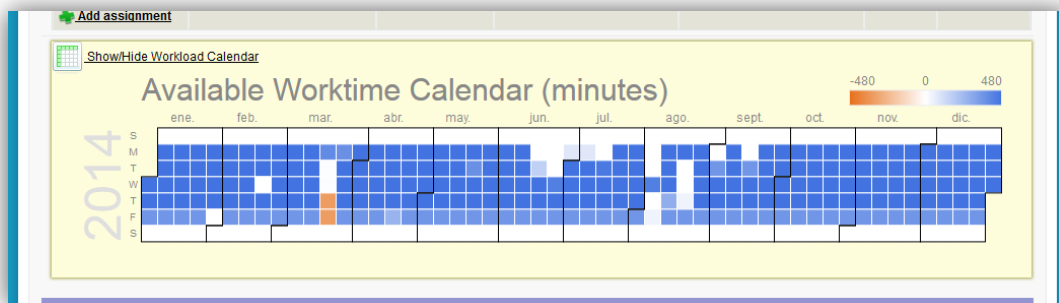


Figura 44. Calendario de cargas de trabajo

Este calendario pertenece a la librería de gráficas de Google para desarrolladores [23], a la cual se puede acceder externamente mediante Javascript, construyendo los parámetros necesarios en el código del controlador.

3.3.3 INFORME DE ACTIVIDADES DE TRABAJO

Uno de los objetivos principales de la herramienta es el informe de las actividades de los empleados. La empresa utiliza una página desarrollada internamente que permite informar, para cada día de la semana, el número de horas que han invertido en cada oportunidad de negocio y en qué tipo de actividad. Este esquema no permite saber el tiempo invertido en cada tarea concreta ni si las estimaciones fueron correctas, y tampoco lleva un seguimiento del coste real de las actividades.

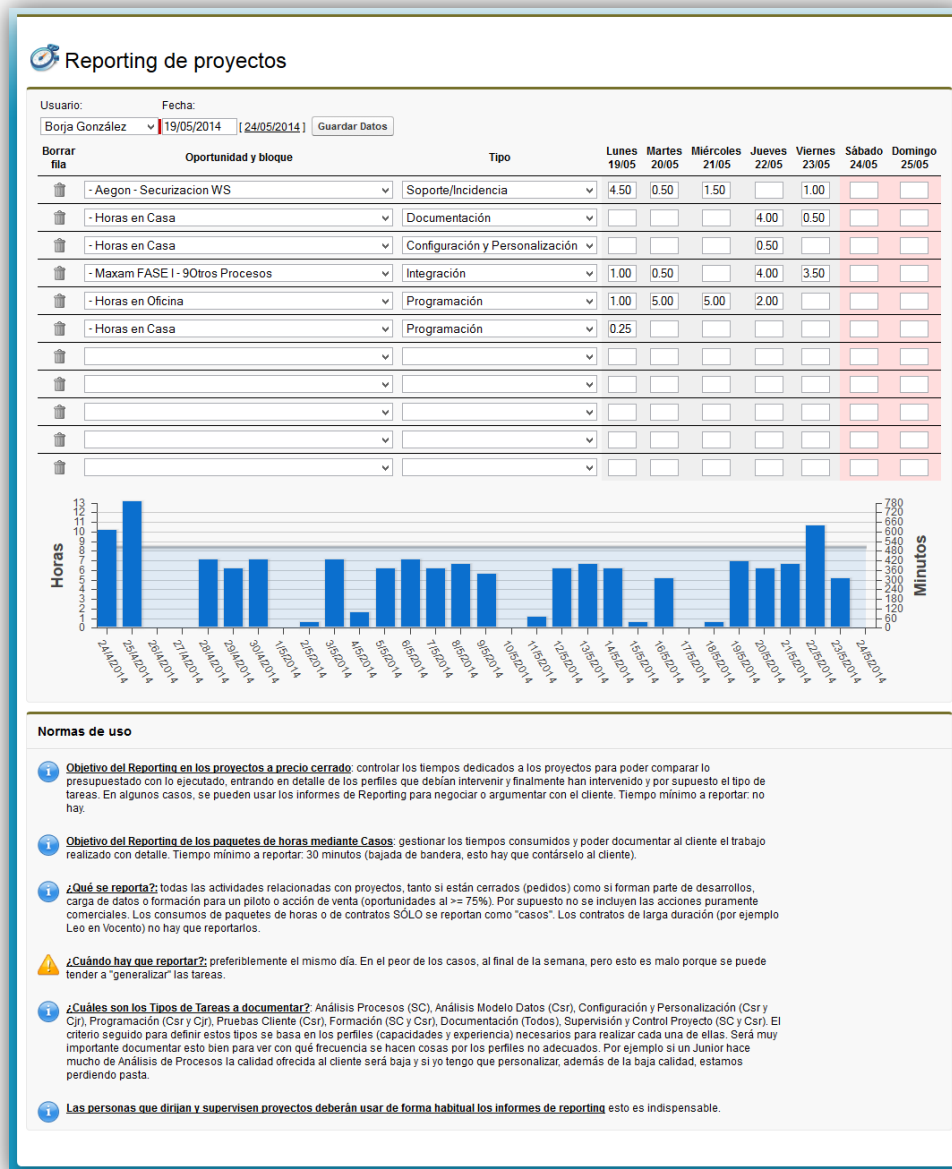


Figura 45. Página de *Reporting* de LeadClic

Por otra parte, la página utilizada es pública y única para todos los empleados, permitiendo a cualquiera que conozca la URL acceder y modificar la información de todos los empleados de la empresa.

La nueva página de informe de actividades es también una página pública, accesible mediante URL. Utiliza los siguientes controladores Apex:

- **ReportingLoginController:** esta clase gestiona todos los flujos de información, tanto en el acceso como en las listas de registros mostradas a continuación.
- **EmailExtension:** esta clase ofrece la funcionalidad de envío de correos electrónicos desde la página.

Para solucionar el problema del libre acceso a la información, la nueva página desarrollada mediante Visualforce cuenta, en primer lugar, con un sistema de autenticación. Para ello se utilizan el correo electrónico y la contraseña almacenados en el registro de empleado de cada usuario, a los que tiene acceso el administrador del sistema.

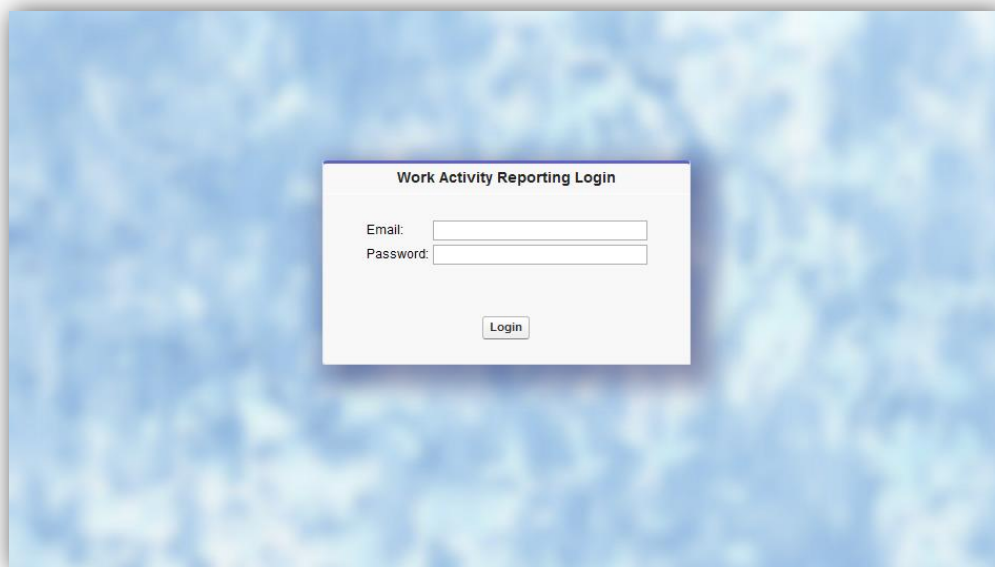


Figura 46. Acceso seguro a página de informe de actividades

El atributo *forceSSL* en el formulario asegura que los datos se enviarán a través de un túnel seguro SSL. Además, las contraseñas de los usuarios se guardan en un campo encriptado en Salesforce.

Se ha programado la página de manera que no sean necesarios saltos a otras páginas para mostrar la información. Si el acceso es correcto, el controlador accederá a la base de datos y cargará la vista de informes. De esta manera, se ahorra tiempo en la transición y hace innecesario el uso de *cookies* u otros métodos para mantener la sesión del usuario.

Si el acceso es correcto se mostrará la lista asignaciones de tareas en las que participa el usuario, agrupadas por tarea y proyecto, y siempre que estén incompletas, o la fecha de finalización no se haya alcanzado, y el proyecto al que pertenecen no esté cerrado.

Assigned Tasks

Test Project

Desarrollo de herramienta de TFG

Desarrollo páginas VisualForce - Handler: Felipe Echanique [Contact]

Página de análisis de proyecto - Due date: 27/06/2014 0:18

Assignment completion %: 70.0 0% Update

Work Activities Date	Time Spent (minutes)	Action
16/06/2014	480	
17/06/2014	200	
18/06/2014	330	
19/06/2014	60	
25/06/2014	150	
Register new work activity		

Desarrollo controladores de páginas - Handler: Borja González [Contact]

Desarrollo controlador reporting de empleados - Due date: 27/06/2014 0:21

Assignment completion %: 0 0% Update

Work Activities Date	Time Spent (minutes)	Action
Register new work activity		

Desarrollo controlador carga de trabajo - Due date: 4/07/2014 0:21

Desarrollo controlador calendario - Due date: 11/07/2014 0:21

Desarrollo controlador análisis proyectos - Due date: 25/07/2014 0:21

Desarrollo clases utilidades - Handler: Borja González [Contact]

Desarrollo Triggers - Handler: Borja González [Contact]

Corrección de errores - Handler: Borja González [Contact]

Implementación - Handler: Borja González [Contact]

Figura 47. Página de informe de actividades

Para cada tarea se facilita el envío de correos electrónicos al responsable, abriendo una ventana en la propia página y haciendo así innecesario abandonarla para comunicarse con el resto del equipo (Figura 41. Envío de correo sin abandonar interfaz).

Desde esta página, el usuario que ha accedido puede registrar nuevas actividades de trabajo en la asignación a la que corresponden, eliminar otras actividades, y establecer el nivel de completitud de cada asignación, ya que no está necesariamente relacionado con el tiempo invertido en la misma.

La información introducida a través de este sistema por los empleados alimentará todos los cálculos de los registros superiores en la jerarquía, y serán fácilmente analizables desde la interfaz de análisis de proyectos.

3.3.4 ANÁLISIS DE CARGAS DE TRABAJO DE EMPLEADOS

A petición de la empresa, se ha incorporado en la herramienta una página de análisis de cargas de trabajo de empleados. Esta página, desarrollada mediante Visualforce, utiliza los siguientes controladores Apex:

- **ResourcesWorkloadController:** esta clase gestiona el acceso a la información y el estado de la página.
- **EmailExtension:** esta clase ofrece la funcionalidad de envío de correos electrónicos desde la página.

La página ha sido asignada a la última de las pestañas de la interfaz principal para facilitar el acceso a la misma. Está compuesta por una lista de los empleados de la empresa, incluyendo para cada uno un calendario de cargas de trabajo y una tabla de asignaciones de tareas con la información elemental.

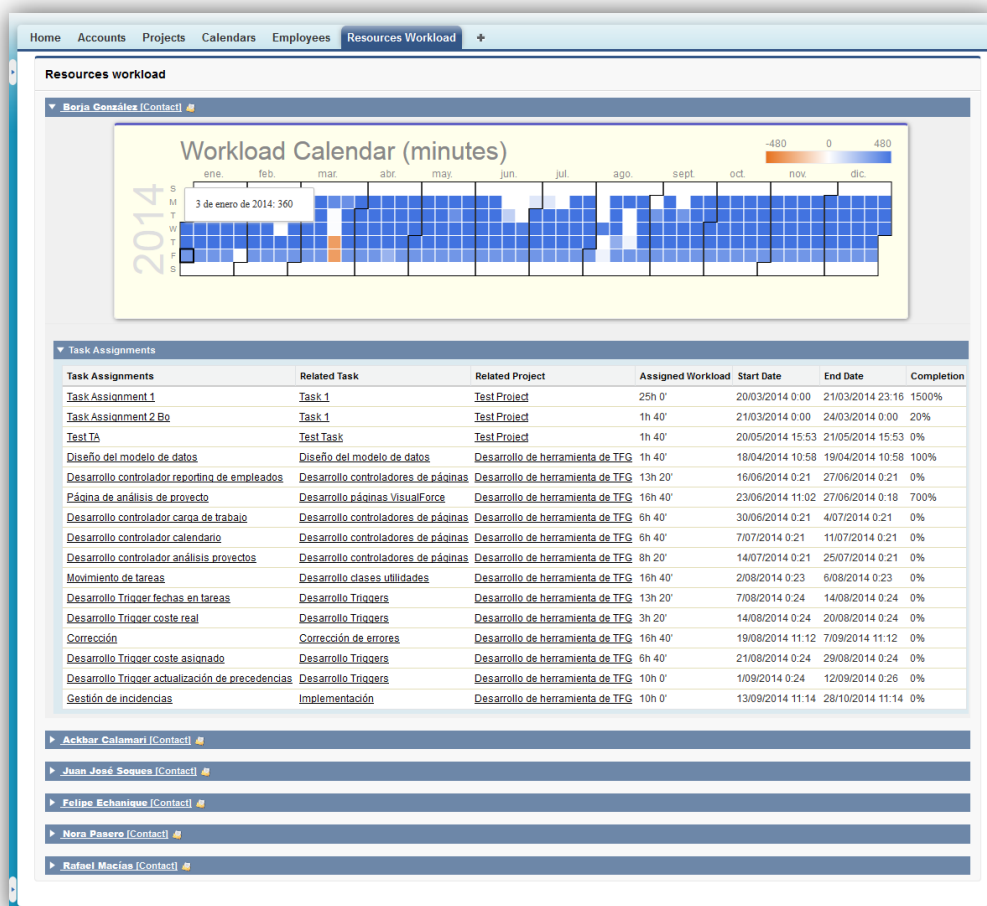


Figura 48. Análisis de cargas de trabajo de empleados

Todas las referencias a registros de la base de datos son enlaces a la página de su registro en el sistema, y se ha implementado un sistema de envío de correos a los empleados mediante el enlace que aparece junto a su nombre. El envío del correo se realiza mediante una interfaz superpuesta a la anterior para evitar la necesidad de tener que salir de la página en la que se tomó la decisión de enviarlo (Figura 41. Envío de correo sin abandonar interfaz).

Se ha implementado una gráfica en la que se muestra sobre un calendario la carga de trabajo disponible de cada empleado. Se representa mediante un código de color la disponibilidad para cada fecha (azul más intenso, mayor disponibilidad; rojo más intenso, menor disponibilidad). Además, se puede conocer la carga exacta (medida en minutos) en cada fecha al poner el cursor sobre la casilla. Para realizar este cálculo se tienen en cuenta las asignaciones del empleado en todos los proyectos de la empresa, lo que es adecuado para un equipo que se movilice entre varios proyectos según las necesidades del momento.

Este calendario pertenece a la librería de gráficas de Google para desarrolladores [23], a la cual se puede acceder externamente mediante Javascript, construyendo los parámetros necesarios en el código del controlador.

3.4 LÓGICA INTERNA PROGRAMADA

En Salesforce, hay dos formas de implementar la lógica interna de una aplicación: mediante configuración, y mediante desarrollo.

La primera de las opciones es más rápida, fiable, sencilla, fácil de mantener y no requiere conocimientos en programación, por lo que es mejor siempre que sea posible. La utilidad principal son los flujos de trabajo, que basados en cambios en la base de datos pueden provocar otros cambios calculados, envíos de correos electrónicos, generación de tareas para usuarios o envíos e mensajes a servicios *web*.

La segunda opción es contraria a la anterior en todos los puntos, pero por contrapartida ofrece una potencia y posibilidades muy superiores. En general, está basada en la programación de clases de Apex, un lenguaje similar a Java que integra procedimientos de la base de datos y métodos y características propios. Por otro lado, están la programación Javascript de botones integrados en la interfaz estándar y en páginas personalizadas, o la gestión de servicios *web* que añadan funcionalidades externas, aunque son algo más difíciles de integrar con el sistema.

En el desarrollo de esta herramienta no ha sido posible utilizar flujos de trabajo, ya sea por no ser suficientemente potentes para las operaciones requeridas o porque no pueden ejecutarse en base a la eliminación de registros. Por tanto, toda la lógica de la aplicación es Apex, distribuida entre clases controladoras de páginas, clases de utilidades y disparadores del sistema, con algo de Javascript para realizar llamadas a las clases o a librerías externas.

3.4.1 DISPARADORES

AssignedCost (TaskAssignment)

Mediante este disparador, que se ejecuta tras modificar el coste presupuestado de un registro de asignación de tarea, se actualizan todas las tareas relacionadas con la asignación recalculando sus costes presupuestados. Los costes presupuestados de la tarea, bloque y proyecto ascendientes en la jerarquía de la asignación se actualizan automáticamente mediante los campos sumatorios habilitados por las relaciones de tipo maestro-detalle que los enlaza.

RealCost (WorkActivity)

Mediante este disparador, que se ejecuta tras modificar el coste de un registro de actividad de trabajo, se actualizan todas las asignaciones de tareas relacionadas con la actividad recalculando sus costes reales. Los costes reales de la asignación, tarea, bloque y proyecto ascendientes en la jerarquía de la actividad se actualizan automáticamente mediante los campos sumatorios habilitados por las relaciones de tipo maestro-detalle que los enlaza.

UpdateCosts (Employee)

Mediante este disparador, que se ejecuta tras modificar el coste salarial de un registro de empleado, se actualizan todas las asignaciones de tareas, y sus respectivas tareas, del empleado.

Invocando un método de la clase auxiliar de utilidades **PMUtilities** se actualizan los costes reales de las asignaciones y los costes presupuestados de la tarea, actualizándose el resto de costes presupuestados y reales de tarea, bloque y proyecto ascendientes en la jerarquía de la asignación automáticamente mediante los campos sumatorios habilitados por las relaciones de tipo maestro-detalle que los enlaza.

PrecedencesUpdate (TaskPrecedence)

Mediante este disparador, que se ejecuta tras insertar un nuevo registro de precedencia de tareas, o modificar el desfase de uno ya existente, se actualiza la tarea dependiente, o la primera tarea del bloque dependiente.

En el primer caso, tan sólo es necesario recalcular la fecha de comienzo de la tarea dependiente, actualizándose las tareas o bloques dependientes de esta última al ejecutarse el disparador **TasksAndBlocksPrecedences**.

En el segundo caso, se calcula el tiempo que se debe desplazar el bloque dependiente, según la relación de precedencia se haya insertado o modificado, y se invocan un método de la clase auxiliar de utilidades **PMUtilities** para mover el bloque completo, que desplaza todas las tareas contenidas en el mismo. Al igual que en el caso anterior, las tareas o bloques dependientes del bloque modificado, o de alguna de sus tareas, se actualizan al ejecutarse el disparador **TasksAndBlocksPrecedences**.

TasksAndBlocksPrecedences (Task)

Mediante este disparador, que se ejecuta tras eliminar o modificar un registro de tarea, se eliminan las relaciones de precedencia en que estaba implicada, o se actualizan todos los bloques y tareas que dependen de la primera si es modificada su fecha de finalización, respectivamente.

En el caso de una modificación, se tiene en cuenta si la entidad dependiente es una tarea, que será desplazada de igual manera que la tarea modificada, o si es un bloque, en cuyo caso es necesario desplazar todas las tareas contenidas en él. Además, en el caso en que una tarea dependa de varias tareas o bloques, incluyendo la dependencia del bloque en el que está contenida, se considerará como válida la relación en la que la tarea se ejecutaría más tarde.

3.4.2 CONTROLADORES DE PÁGINAS

CalendarAllProjectsController

Esta clase se utiliza como controlador para la página de calendario en la que se muestran los eventos de todos los proyectos, así como las excepciones del calendario definido como estándar. Esta página se muestra desde un enlace en la barra lateral de Salesforce.

El controlador se encarga de acceder a la base de datos al cargarse la página, obteniendo los registros de proyectos, bloques de tareas, tareas y excepciones del calendario por defecto, y construyendo a partir de ellos los objetos tipo evento que luego la página recorrerá para poblar el calendario. Por otra parte, incluye el método necesario para actualizar la base de datos cuando se realiza algún cambio desde la página arrastrando un evento. Si se arrastra un bloque de tareas, el controlador invoca un método de la clase auxiliar de utilidades **PMUtilities** para mover el bloque completo, que desplaza todas las tareas contenidas en el mismo.

CalendarEmployeeExtension

Esta clase se utiliza como controlador para la página de calendario en la que se muestran sus asignaciones de tareas y excepciones incluidas en su calendario asignado, o registradas como personales. Es accesible desde las páginas de registros de empleados, mediante el botón a ese efecto.

El controlador se encarga de acceder a la base de datos al cargarse la página, obteniendo los registros de asignaciones de tareas y excepciones de calendario, y construyendo a partir de ellos los objetos tipo evento que luego la página recorrerá para poblar el calendario. Por otra parte, incluye el método necesario para actualizar la base de datos cuando se realiza algún cambio desde la página arrastrando un evento personal del empleado.

CalendarExtension

Esta clase se utiliza como controlador para la página de calendario en la que se muestran las excepciones incluidas en un registro de calendario. Es accesible desde las páginas de registros de calendarios, mediante el botón a ese efecto.

El controlador se encarga de acceder a la base de datos al cargarse la página, obteniendo los registros de excepciones de calendario, y construyendo a partir de ellos los objetos tipo evento que luego la página recorrerá para poblar el calendario. Por otra parte, incluye el método necesario para actualizar la base de datos cuando se realiza algún cambio desde la página arrastrando un evento.

CalendarProjectExtension

Esta clase se utiliza como controlador para la vista de calendario de un proyecto. En él se muestran sus bloques de tareas y tareas, así como las excepciones del calendario por defecto. Es accesible desde las páginas de registros de proyecto, mediante el botón a ese efecto, y también se utiliza para generar la vista de calendario en la página de análisis de proyectos.

El controlador se encarga de acceder a la base de datos, obteniendo los registros de bloques de tarea, tareas y excepciones de calendario, y construyendo a partir de ellos los objetos tipo evento que luego la página recorrerá para poblar el calendario. Por otra parte, incluye el método necesario para actualizar la base de datos cuando se realiza algún cambio desde la página arrastrando un evento. Si se arrastra un bloque de tareas, el controlador invoca un método de la clase auxiliar de utilidades **PMUtilities** para mover el bloque completo, que desplaza todas las tareas contenidas en el mismo.

EmailExtension

Esta clase se utiliza como controlador en la página de envío de correos accesible desde la página de un registro de empleado en el botón a tal efecto, o como extensión en las páginas en las que se desea implementar la posibilidad de enviar correos electrónicos a usuarios, como la página de cargas de trabajo o la de informe de tiempos de trabajo.

Obteniendo los datos de destinatario, remitente, asunto y texto de variables utilizadas como parámetro, construye el objeto de correo electrónico y realiza la llamada interna del sistema.

ProjectAnalysisExtension

Esta clase se utiliza como controlador en la página de análisis de proyectos, proporcionando la gestión de toda la información necesaria y del estado de la página. En el momento de ser cargada la página, el controlador obtiene la información de todas las pestañas para agilizar el proceso posterior de navegación.

Obtiene toda la información relacionada con asignaciones de tarea, tareas, bloques de tareas y recursos humanos y genera los parámetros necesarios para construir las gráficas de completitud, EVM, diagrama de Gantt y cargas de trabajo, de manera que cuando se necesite cargar una sección los valores ya estén preparados. También realiza los cálculos de los parámetros EVM y organiza la jerarquía de tareas para ser mostrada en la pestaña de lista de tareas.

También gestiona el envío de correos desde las pestañas de lista de tareas y cargas de trabajo, recogiendo los parámetros de la interfaz y realizando las llamadas al sistema.

ProManHomeController

Esta clase se utiliza como controlador del marco mostrado en la pantalla de *Home*, y simplemente accede a la base de datos para obtener una lista de los proyectos sin cerrar con los campos mostrados en la interfaz y ordenados por fecha de finalización.

ReportingLoginController

Esta clase se utiliza como controlador de la página de informe de actividades de trabajo de los empleados. Valida las credenciales introducidas por el usuario para acceder al sistema, y accede a la base de datos para obtener sus asignaciones de tareas, así como las tareas, bloques de tarea y proyectos relacionados para mostrarlos en la jerarquía de la interfaz.

También proporciona los métodos necesarios para eliminar y registrar actividades de trabajo en cada asignación.

ResourcesWorkloadController

Esta clase se utiliza como controlador en la página de cargas de trabajo de empleados, y como extensión en la página de análisis de proyectos y se encarga de generar los parámetros necesarios para construir las gráficas de cargas de trabajo de empleados.

Al cargarse, la clase obtiene de la base de datos información sobre los calendarios de trabajo y empleados, y genera una lista y un mapa clases contenedoras de empleados con calendarios para acceder a ellos desde la página de cargas de trabajo y análisis de proyecto, respectivamente.

Para cada fecha a mostrar en el calendario de un empleado, la clase tiene en cuenta el tiempo de trabajo para ese día de la semana según su calendario, las excepciones del calendario por defecto y el calendario del empleado, y las asignaciones de tarea.

Cada asignación se distribuye de manera que pueda completarse lo antes posible, llenando los primeros días desde su fecha de comienzo hasta que se haya distribuido toda su carga de trabajo. En esta primera pasada se anota el recuento de días de trabajo cuyo día de la semana tiene algo de tiempo de trabajo y que no coinciden con excepciones de calendario.

Si al finalizar este proceso sigue quedando carga sin asignar, se hace una división del tiempo restante entre el número de días de trabajo calculado (teniendo en cuenta divisiones no exactas) y se distribuye la carga, dando lugar a sobrecarga para el empleado en esas fechas, lo que se mostrará con tonalidades rojas.

3.4.3 CLASE DE UTILIDADES

Aparte de las clases desarrolladas para cada objetivo específico, se ha creado una clase de utilidades, **PMUtilities**, que proporciona métodos llamados desde clases externas para evitar la replicación de código o proporcionar inteligencia en el servidor a botones Javascript.

Esta clase contiene los siguientes métodos:

- **calculateBaseTAEnd**: método que calcula la fecha de finalización prevista de una asignación de tarea teniendo en cuenta su fecha de comienzo y carga de trabajo, la carga de trabajo del empleado, sus tiempos de trabajo cada día de la semana y las excepciones de calendario, si no se incurriera en sobrecarga para ninguna fecha. Este método se declara como servicio *web* para poder ser invocado desde botones Javascript.
- **moveTaskBlock**: método que desplaza un bloque de tareas completo, desplazando todas las tareas contenidas en él el periodo de tiempo especificado por parámetro.
- **updateCosts**: método que recalcula para una tarea su coste presupuestado a partir de sus asignaciones, y los costes reales de las asignaciones a partir de los registros de actividades de trabajo relacionados. Este método se declara como servicio *web* para poder ser invocado desde botones Javascript.

4.PRUEBAS CON USUARIOS

Tras finalizar el desarrollo de la herramienta se llevaron a cabo pruebas de usabilidad con dos usuarios finales de la empresa. El perfil de los dos es de gestores de la empresa y de proyectos, con conocimientos avanzados en Salesforce a nivel de usuario final, pero con conocimientos muy limitados en el aspecto técnico de desarrollo Apex y Visualforce. Dado que el público objetivo va a tener conocimientos avanzados en Salesforce, la muestra es adecuada.

En cada prueba se informó al usuario de la estructura del modelo de datos, y a continuación se planteó un escenario típico de uso en el que se debía crear un proyecto, bloques, tareas, dependencias y asignaciones, informar de actividades de trabajo y analizar toda la información obtenida para un proyecto. Por último, se mostró la herramienta de manera más detallada, anotando posibles modificaciones y mejoras a modo de validación. Cada prueba duró aproximadamente veinte minutos y fue grabada para su posterior análisis.

Por último, al finalizar las pruebas, se entregó a los dos usuarios un cuestionario de satisfacción (ANEXO 1. CUESTIONARIO DE USABILIDAD) con preguntas sobre la impresión general del producto y una encuesta tipo AttarkDiff [26]. Los resultados se muestran en la siguiente tabla.

Tras las pruebas, se han identificado como puntos débiles en la usabilidad la vista del calendario, ya que no permite una visión global del proyecto y los eventos saturan el gráfico y la ausencia de filtros en algunas relaciones que eviten errores y algunos nombres.

Como puntos fuertes cabe destacar la página de análisis de proyecto, en concreto las vistas de completitud, diagrama de Gantt y recursos involucrados con sus cargas de trabajo. También es destacable la página de informe de actividades de empleados, que sigue un esquema superior a la utilizada actualmente.

	U1	U2	Media
El producto permite realizar las tareas solicitadas de forma fácil	5	3	4
La navegación entre las opciones resulta clara	4	4	4
La apariencia general del producto es agradable	4	5	4.5
Las imágenes que aparecen en la interfaz del producto son representativas	5	4	4.5
El texto que aparece en la interfaz del producto es relevante	4	3	3.5
La interfaz de usuario tiene una estructura y organización claras	4	5	4.5
Los nombres de las opciones me han parecido claros y representativos	4	2	3
Me ha sido fácil encontrar la información que buscaba	4	4	4
El producto es adecuado para las necesidades de distintos tipos de usuarios	4	5	4.5
En general el producto me parece fácil de usar	4	4	4
Me gustaría usar este producto a menudo	3	4	3.5
La mayoría de las personas pueden aprender rápidamente a usarlo	3	5	4

Tabla 1. Impresión general del producto (puntuación de 1 a 5)

	U1	U2	Media
Tecnológico / Humano	4	2	3
Poco práctico / Muy práctico	5	7	6
Poco predecible / Predecible	5	6	5.5
Confuso / Claro	5	6	5.5
Poco profesional / Muy profesional	6	5	5.5
Vulgar / Estético	5	6	5.5
Convencional / Original	5	3	4
Banal / Creativo	6	5	5.5
Ortodoxo / Atrevido	4	2	3
Conservador / Innovador	4	3	3.5
Aburrido / Interesante	6	6	6
Desagradable / Agradable	6	6	6
Feo / Bonito	5	5	5
Ingrato / Grato	6	6	6
Provoca rechazo / Provoca aceptación	5	5	5
Desanima / Motiva	4	6	5

Tabla 2. Encuesta AttrakDiff (puntuación de 1 a 7)

5.RESULTADOS Y CONCLUSIONES

El objetivo principal de esta herramienta era integrar las soluciones a muchas de las necesidades de la empresa, que actualmente se hayan dispersas entre multitud de aplicaciones, de una manera eficaz, robusta y acorde a los requisitos planteados.

El objetivo se ha cumplido en gran medida, ya que el resultado de este desarrollo es una aplicación basada en la nube, y por tanto portable, con un modelo de datos potente y muy eficiente de gestión de proyectos, una interfaz elaborada y usable y una lógica interna sólida que garantiza la calidad de los datos y la robustez en el uso.

De los planes e ideas que implementar en la herramienta al comienzo del proyecto, la mayoría se han podido llevar a cabo. Los principales son los siguientes:

- La automatización de la mayoría de procesos internos mediante un diseño eficiente de la base de datos y la programación de disparadores y controladores
- La integración con el CRM mediante la relación entre proyectos y oportunidades de negocio
- La planificación de proyectos y su posterior seguimiento, mostrando valoraciones de estado y gráficas de completitud, EVM, diagrama de Gantt y calendario para facilitar la toma de decisiones correctivas
- La gestión de recursos humanos, teniendo en cuenta costes salariales, horarios, calendarios y cargas de trabajo
- La utilización de interfaces accesibles y navegables minimizando los saltos de página
- La mejora de la página de informe de actividades de trabajo de empleados, obteniendo la información a nivel de asignaciones de tarea, y protegiendo el portal mediante un acceso seguro
- La mejora de la comunicación interna en los equipos mediante el acceso rápido al envío de correos electrónicos desde otras interfaces
- La realización de pruebas con usuarios finales para validar el trabajo completado

No se ha llegado a implementar, sin embargo, la posibilidad de modificar los datos de un proyecto desde el diagrama de Gantt, o una interfaz más amigable para gestionar relaciones de precedencia entre tareas y bloques.

La herramienta ha resultado satisfactoria para los miembros de la empresa en muchos aspectos, destacando la interfaz de análisis de proyectos la página de informe de actividades de trabajo. Con un mayor desarrollo y perfeccionamiento, podría llegar a cambiar la metodología de trabajo en la empresa y llevarla a un nivel superior, preparado para su crecimiento.

ANEXO 1. CUESTIONARIO DE USABILIDAD

Cuestionario de satisfacción

Datos personales

Participante:

Fecha:

Impresión general del producto

Valora de 1 a 5 las siguientes afirmaciones, donde 1 significa “estoy en completo desacuerdo” y 5 significa “estoy totalmente de acuerdo”.

	1	2	3	4	5
El producto permite realizar las tareas solicitadas de forma fácil					
La navegación entre las opciones resulta clara					
La apariencia general del producto es agradable					
Las imágenes que aparecen en la interfaz del producto son representativas					
El texto que aparece en la interfaz del producto es relevante					
La interfaz de usuario tiene una estructura y organización claras					
Los nombres de las opciones me han parecido claros y representativos					
Me ha sido fácil encontrar la información que buscaba					
El producto es adecuado para las necesidades de distintos tipos de usuarios					
En general el producto me parece fácil de usar					
Me gustaría usar este producto a menudo					
La mayoría de las personas pueden aprender rápidamente a usarlo					

Encuesta Attrak-Diff

Para cada pareja de términos, marca la posición que mejor indique tu opinión respecto al producto utilizado.

Tecnológico	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Humano
Poco práctico	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Muy práctico
Poco predecible	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Predecible
Confuso	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Claro
Poco profesional	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Muy profesional
Vulgar	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Estético
Convencional	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Original
Banal	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Creativo
Ortodoxo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Atrevido
Conservador	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Innovador
Aburrido	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Interesante
Desagradable	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Agradable
Feo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Bonito
Ingrato	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Grato
Provoca rechazo	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Provoca aceptación
Desanima	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Motiva

ANEXO 2. CÓDIGO

Como referencia, se incluye el código de dos clases representativas de la herramienta, debidamente comentadas.

PMUtilities

```
// Borja González Orduna
// 19-02-2014

// Class with common task methods
global class PMUtilities {

    // Method for calculating and updating a task assignment's end date based on its workload
    and the employee's schedule
    webService static void calculateBaseTAEnd(Id taID) {

        // Task assignment to be handled
        TaskAssignment__c tAssigned = [SELECT Id, Employee__c, Employee__r.Calendar__r.Id,
Task__c, AssignedWorkload__c,
        Employee__r.Calendar__r.Monday__c,
        Employee__r.Calendar__r.Tuesday__c, Employee__r.Calendar__r.Wednesday__c,
        Employee__r.Calendar__r.Thursday__c, Employee__r.Calendar__r.Friday__c,
        Employee__r.Calendar__r.Saturday__c, Employee__r.Calendar__r.Sunday__c,
        Task__r.StartDate__c, Task__r.EndDate__c
        FROM TaskAssignment__c
        WHERE Id = :taID];

        // List of the employee's calendar exceptions inside the assignment's period of time
        List<CalendarException__c> empAbscences = [SELECT StartDate__c, EndDate__c
        FROM CalendarException__c
        WHERE (Employee__c = :tAssigned.Employee__c
        OR Calendar__c =
:tAssigned.Employee__r.Calendar__r.Id)
        AND EndDate__c >
:tAssigned.Task__r.StartDate__c];

        // List of the employee's task assignments overlapping the assignment's period of time
        List<TaskAssignment__c> empAssignments = [SELECT Id, StartDate__c, EndDate__c
        FROM TaskAssignment__c
        WHERE Employee__c = :tAssigned.Employee__c
        AND EndDate__c >
:tAssigned.Task__r.StartDate__c
        AND Id != :tAssigned.Id];

        // List of time lapses when the employee won't be available for this assignment
        List<TimeLapse> NATime = new List<TimeLapse>();
        for(CalendarException__c ea : empAbscences) {
            NATime.add(new TimeLapse(ea.StartDate__c, ea.EndDate__c));
        }
        for(TaskAssignment__c ea : empAssignments) {
            NATime.add(new TimeLapse(ea.StartDate__c, ea.EndDate__c));
        }

        // List of the employee's working times per day of the week
        List<Integer> workTimes = new List<Integer>{
            tAssigned.Employee__r.Calendar__r.Monday__c.intValue(),
            tAssigned.Employee__r.Calendar__r.Tuesday__c.intValue(),
            tAssigned.Employee__r.Calendar__r.Wednesday__c.intValue(),
            tAssigned.Employee__r.Calendar__r.Thursday__c.intValue(),
            tAssigned.Employee__r.Calendar__r.Friday__c.intValue(),
            tAssigned.Employee__r.Calendar__r.Saturday__c.intValue(),
            tAssigned.Employee__r.Calendar__r.Sunday__c.intValue()
        }
    }
}
```



```

};

// Task assignment workload time yet to be counted
Integer wtLeft = tAssigned.AssignedWorkload__c.intValue();
// Number of days to add to the task start date
Integer daysToAdd = 0;
// Whether current date must not be counted
Boolean leapDate;

// Whether the task assignment's start date has already been set
Boolean sdSet = false;
// By default, the task assignment's start date is the same as the task's
tAssigned.BaseStartDate__c = tAssigned.Task__r.StartDate__c;
// Current day of the week while counting
Integer weekDay =
tAssigned.Task__r.StartDate__c.date().toStartOfWeek().daysBetween(tAssigned.Task__r.StartDate__c.date());
// Days are looped over until all the working time has been counted
// d represents how many days from the task start date have been looped over
for(Integer d=0; wtLeft > 0; d++) {

    // Current day is counted only if it has any worktime assigned or if it is not
    // contained in the list of not
    // available time lapses
    // If the assignment's base start date has not been set yet, that's the date to
    use
    // Otherwise, the assignment's base end date is the one to use
    leapDate = false;
    if(workTimes[weekDay] == 0) {
        leapDate = true;
    } else {
        for(Integer natl=0; natl<NATime.size() && !leapDate; natl++) {
            if(!sdSet && NATime[natl].containsDT(tAssigned.BaseStartDate__c)) ||
            (sdSet && NATime[natl].containsDT(tAssigned.BaseEndDate__c))) {
                leapDate = true;
            }
        }
    }

    // If current day is a leap date and the task assignment's start date has not been
    // set yet, it is delayed one
    // more day. If the task assignment's start date has been set, the base end date
    // is delayed one more day
    if(leapDate) {
        if(!sdSet) {
            tAssigned.BaseStartDate__c = tAssigned.BaseStartDate__c.addDays(1);
        } else {
            tAssigned.BaseEndDate__c = tAssigned.BaseEndDate__c.addDays(1);
        }
    }
    // Otherwise, its working time should be subtracted from the left workload
    } else {
        // If the task assignment's start date had not been set yet, the changed is
        // now registered and the end date
        // is set to the start date as default
        if(!sdSet) {
            sdSet = true;
            tAssigned.BaseEndDate__c = tAssigned.BaseStartDate__c;
        }
        // When there is less time workload to be counted than working time in the
        // current week day, only the former
        // is counted
        if(wtLeft < workTimes[weekDay]) {
            tAssigned.BaseEndDate__c = tAssigned.BaseEndDate__c.addMinutes(wtLeft);
        }
        // Otherwise, a whole day is counted
        } else {
            tAssigned.BaseEndDate__c = tAssigned.BaseEndDate__c.addDays(1);
        }
    }
}

```

```

        // Afterwards, the current week day working time is subtracted from the left
workload
        wtLeft -= workTimes[weekDay];
    }
    // Current week day advances by one, starting back at monday after sunday
    weekDay = Math.mod(weekDay+1, 7);
}

    update tAssigned;
}

    // Method for moving a whole task block, that is, moving each of its contained tasks
    public void moveTaskBlock(Id blockId, Integer dayDelta, Integer minuteDelta) {

        List<Task__c> blockTasks = [SELECT Id, StartDate__c, EndDate__c FROM Task__c WHERE
TaskBlock__c = :blockId];

        for(Task__c nTask : blockTasks) {
            nTask.StartDate__c = nTask.StartDate__c.addDays(Integer.valueOf(dayDelta));
            nTask.StartDate__c = nTask.StartDate__c.addMinutes(Integer.valueOf(minuteDelta));

            nTask.EndDate__c = nTask.EndDate__c.addDays(Integer.valueOf(dayDelta));
            nTask.EndDate__c = nTask.EndDate__c.addMinutes(Integer.valueOf(minuteDelta));
        }

        update blockTasks;
    }

    // Method for updating a task's and its assignments' assigned and real cost based on all
    of their assignments and related activities
    webservice static void updateCosts(Id taskID) {

        // Work activities real costs grouped by task assignment
        AggregateResult[] wacsCosts = [SELECT TaskAssignment__r.Id taID, SUM(ActivityCost__c)
realCost
                                     FROM WorkActivity__c WHERE TaskAssignment__r.Task__r.Id
= :taskID
                                     GROUP BY TaskAssignment__r.Id];

        // List of task assignments to update
        List<TaskAssignment__c> tAssignments = new List<TaskAssignment__c>();
        for(AggregateResult wac : wacsCosts) {
            tAssignments.add(new TaskAssignment__c(Id = (Id)wac.get('taID'), RealCost__c =
(Decimal)wac.get('realCost')));
        }

        // Task assignments' costs grouped by task
        AggregateResult[] tasCosts = [SELECT Task__r.Id, SUM(EstimatedCost__c) estCost
                                     FROM TaskAssignment__c WHERE Task__r.Id = :taskID
                                     GROUP BY Task__r.Id];

        // List of tasks to update
        List<Task__c> tasks = new List<Task__c>();
        for(AggregateResult tac : tasCosts) {
            tasks.add(new Task__c(Id = taskID, AssignedCost__c =
(Decimal)tac.get('estCost')));
        }

        update tAssignments;
        update tasks;
    }

    // Wrapper class for easier time lapses management
    public class TimeLapse {
        public DateTime StartDate {get;set;}
    }

```

```

        public DateTime EndDate {get;set;}

        // Constructor
        public TimeLapse(DateTime st, DateTime et) {
            StartDate = st;
            EndDate = et;
        }

        // Checks whether an specific DateTime is contained in the time lapse
        public Boolean containsDT(DateTime dt) {
            return (StartDate<dt && EndDate>dt);
        }
    }
}

```

ProjectAnalysisExtension

```

// Borja González Orduna
// 11-3-2014

// Controller class for the project analysis page
public class ProjectAnalysisExtension {

    // Analysed Project
    public Project__c project {get;set;}

    // View tab index
    public Integer viewIndex {get;set;}

    // ----- Completion tab -----

    //      Workload completion

    // Completion chart
    public String chFields {get;set;}
    public String chValues {get;set;}
    public String chOptions {get;set;}
    // EVM chart
    public String evFields {get;set;}
    public String evValues {get;set;}
    public String evOptions {get;set;}
    // Completion status
    public Integer scheduledCompletion {get;set;}
    public Integer realCompletion {get;set;}

    //      EVM

    // Budgeted Cost for Work Scheduled
    public Decimal BCWS {get;set;}
    // Budgeted Cost for Work Performed
    public Decimal BCWP {get;set;}
    // Actual Cost of Work Performed
    public Decimal ACWP {get;set;}

    // Variances

    public Decimal CostVar {get;set;}
    public Decimal SchedVar {get;set;}
    public Decimal VarAtComp {get;set;}
}

```

```

// ----- Tasks List and Resources tabs -----

public List<WrappedProjectDate> projectDates {get;set;}
public List<Employee__c> resources {get;set;}
public List<Task__c> tasks {get;set;}
public Task__c newTask {get;set;}
public Id newTBlockID {get;set;}
public List<WrappedBlock> tBlocks {get;set;}
public TaskBlock__c newBlock {get;set;}
public Boolean blockInput {get;set;}

public TaskAssignment__c newTaskAssignment {get;set;}
public Id newTAEmpID{get;set;}

// ----- Emailing Layout -----

public Boolean emailDisplayed {get;set;}
public String emailAddress {get;set;}
public String emailRecipient {get;set;}
public String emailSubject {get;set;}
public String emailText {get;set;}

// ----- Gantt Diagram -----

// Data json string
public String jsonString {get;set;}

// ***** Class constructor *****

public ProjectAnalysisExtension(ApexPages.StandardController proj) {

    project = (Project__c)proj.getRecord();

    // When loaded, no tab will be displayed
    viewIndex = 0;
    // Email interface is not displayed
    emailDisplayed = false;
    // Tasks data is retrieved
    resetTasksData();
    // Resources data is retrieved
    resetResourcesData();

    // A list of wrapped dates is composed for charts drawing
    projectDates = new List<WrappedProjectDate>();
    // Completion / EVM data is composed
    composeCompletion();
    // Gantt data is composed in a json string
    jsonString = prepareData();

}

// ***** Completion / EVM *****

// Method for loading or updating completion / EVM data
public void composeCompletion() {
    calcProjectMetrics();
    resetCharts();
}

// Populates the list of project dates and calculates metrics
public void calcProjectMetrics() {

    scheduledCompletion = 0;
    realCompletion = 0;
    BCWS = 0;

```

```

BCWP = 0;
ACWP = 0;

projectDates.clear();
for(Task__c t : tasks) {

    // Workload is 0 at project start
    if(projectDates.size() == 0) {
        projectDates.add(new WrappedProjectDate(
            t.EndDate__c.date(), t.IsCompleted__c, t.Workload__c.intValue(),
            t.AssignedWorkload__c.intValue(), t.Completion__c.intValue(),
            t.AssignedCost__c, t.IsCompleted__c? t.AssignedCost__c : 0,
            t.IsCompleted__c? t.RealCost__c : 0));
    }
    // If this task's end date has already been used, its previous record is modified
    if(projectDates[projectDates.size()-1].projDate < t.EndDate__c.date()) {
        projectDates.add(new WrappedProjectDate(
            t.EndDate__c.date(), t.IsCompleted__c, t.Workload__c.intValue(),
            t.AssignedWorkload__c.intValue(), t.Completion__c.intValue(),
            t.AssignedCost__c, t.IsCompleted__c? t.AssignedCost__c : 0,
            t.IsCompleted__c? t.RealCost__c : 0));
    }
    // Else, a new record is created
    } else {
        projectDates[projectDates.size()-1].ScheduledWorkload +=
        t.Workload__c.intValue();
        projectDates[projectDates.size()-1].AssignedWorkload +=
        t.AssignedWorkload__c.intValue();
        // EVM specifies only completed tasks account for completion
        if(t.IsCompleted__c) {
            projectDates[projectDates.size()-1].CompletedWorkload +=
            t.Completion__c.intValue();
        }
        // If this task's end date is previous to or today, its workload values are
        accounted for today's status
        if(t.EndDate__c <= System.today()) {
            scheduledCompletion += t.Workload__c.intValue();
            realCompletion += t.Completion__c.intValue();
            BCWS += (t.AssignedCost__c!=null? t.AssignedCost__c : 0);
            // EVM specifies only completed tasks account for work done
            BCWP += (t.IsCompleted__c? (t.AssignedCost__c!=null? t.AssignedCost__c : 0) :
0);
            ACWP += (t.IsCompleted__c? (t.RealCost__c!=null? t.RealCost__c : 0) : 0);
        }
    }

    BCWS.setScale(2);
    BCWP.setScale(2);
    ACWP.setScale(2);

}

// Method for initializing chart parameters
public PageReference resetCharts() {

    Integer iSeries = 0;
    Integer totalScheduled = 0;
    Integer totalCompleted = 0;
    Integer totalAssigned = 0;

    Decimal totalBCWS = 0;
    Decimal totalBCWP = 0;
    Decimal totalACWP = 0;

    // Completion chart fields
    chFields = '['Date\ ', \'Scheduled Workload\ ', \'Assigned Workload\ ', \'Completed
Workload\ ']' ;

```

```

// EVM chart fields
evFields = '['Date\','BCWS\','BCWP\','ACWP\']';

chValues = ', [new Date(' + project.StartDate__c.year() + ', ' +
(project.StartDate__c.month()-1) + ', ' + project.StartDate__c.day() + '), 0, 0, 0]';
evValues = ', [new Date(' + project.StartDate__c.year() + ', ' +
(project.StartDate__c.month()-1) + ', ' + project.StartDate__c.day() + '), 0, 0, 0]';
for(WrappedProjectDate pd : projectDates) {
    totalScheduled += pd.ScheduledWorkload;
    totalAssigned += pd.AssignedWorkload;
    totalCompleted += pd.CompletedWorkload;
    chValues += ', [new Date(' + pd.projDate.year() + ', ' + (pd.projDate.month()-1) +
', ' + pd.projDate.day() + '), ' + totalScheduled + ', ' + totalAssigned + ', ' +
totalCompleted + ']';

    totalBCWS += pd.BCWS;
    totalBCWP += pd.BCWP;
    totalACWP += pd.ACWP;
    evValues += ', [new Date(' + pd.projDate.year() + ', ' + (pd.projDate.month()-1) +
', ' + pd.projDate.day() + '), ' + totalBCWS + ', ' +
        + (pd.ProjDate<=System.today())? totalBCWP + ', ' + totalACWP : 'null, null' )
+ ']';
}
//chValues += ', [new Date(' + project.EndDate__c.year() + ', ' +
(project.EndDate__c.month()-1) + ', ' + project.EndDate__c.day() + '), ' +
project.TasksWorkload__c + ', ' + project.AssignedWorkload__c + ', ' + project.Completion__c +
']';
evValues += ', [new Date(' + project.EndDate__c.year() + ', ' +
(project.EndDate__c.month()-1) + ', ' + project.EndDate__c.day() + '), ' + totalBCWS + ', ' +
null + ', ' + null + ']';

return null;
}

// ***** Gantt Chart *****

//Temp Method to prepare the Data
private String prepareData() {

    List<tRow> r1 = new List<tRow>();
    tRow r;
    RowDetail obj;

    for(WrappedBlock tb : tBlocks) {

        r = new tRow();
        r.values = new List<RowDetail>();
        r.name = 'Block: ';
        r.desc_PH = '<a href="' + URL.getSalesforceBaseUrl().toExternalForm() + '/' +
tb.TaskBlock.Id + '" style="font-weight: bold;">' + '&darr;' + tb.TaskBlock.Name + ' &darr;'
+ '</a>';

        obj = new RowDetail();
        // Date times in milliseconds in local timezone
        obj.setFromDate(String.valueOf(
            tb.TaskBlock.StartDate__c.getTime() +
UserInfo.getTimeZone().getOffset(tb.TaskBlock.StartDate__c)));
        obj.setToDate(String.valueOf(
            tb.TaskBlock.EndDate__c.getTime() +
UserInfo.getTimeZone().getOffset(tb.TaskBlock.EndDate__c)));
        obj.desc_PH = 'Start Date:&nbsp;' + tb.TaskBlock.StartDate__c.format() +
            '<br> End Date:&nbsp;' + tb.TaskBlock.EndDate__c.format() +
            '<br> Assignments Cost:&nbsp;' + (tb.TaskBlock.AssignedCost__c!=null?
(tb.TaskBlock.AssignedCost__c + ' €') : '') +

```

```

        '</br> Real Cost to date:&emsp;' + (tb.TaskBlock.AssignedCost__c!=null
        && tb.TaskBlock.RealCost__c!=null? (tb.TaskBlock.RealCost__c + ' €') : '');

        obj.label =
        ((100*tb.TaskBlock.Completion__c/tb.TaskBlock.Workload__c).setScale(2)) + ' %';
        obj.customClass = 'ganttGreen';

        r.values.add(obj);
        r1.add(r);

        for(Task__c tk : tb.Tasks) {

            r = new tRow();
            r.values = new List<RowDetail>();
            r.name = 'Task: ';
            r.desc_PH = '<a href="' + URL.getSalesforceBaseUrl().toExternalForm() + '/' +
            tk.Id + '>' + '&rarr;' + tk.Name + '</a>';

            obj = new RowDetail();
            obj.id = tk.Id;
            obj.dep = tb.Tasks[0].id;
            // Date times in milliseconds in local timezone
            obj.setFromDate(String.valueOf(
            tk.StartDate__c.getTime() +
            UserInfo.getTimeZone().getOffset(tk.StartDate__c)));
            obj.setToDate(String.valueOf(
            tk.EndDate__c.getTime() +
            UserInfo.getTimeZone().getOffset(tk.EndDate__c)));

            obj.desc_PH = 'Start Date:&emsp;' + tk.StartDate__c.format() +
            '</br> End Date:&emsp;' + tk.EndDate__c.format() +
            '</br> Assignments Cost:&emsp;' + (tk.AssignedCost__c!=null?
            (tk.AssignedCost__c + ' €') : '') +
            '</br> Real Cost to date:&emsp;' + (tk.AssignedCost__c!=null &&
            tk.RealCost__c!=null? (tk.RealCost__c + ' €') : '');
            obj.label = ((100*tk.Completion__c/tk.Workload__c).setScale(2)) + ' %';
            obj.customClass = 'ganttOrange';

            r.values.add(obj);
            r1.add(r);

        }

    }

    return JSON.serialize(r1).replaceAll('frm_PH','from').replaceAll('desc_PH','desc');
}

// ***** Tasks List *****

// Method for retrieving all tasks related data
public void resetTasksData() {

    tasks = [SELECT Id, Name, Handler__c, Handler__r.Name, Handler__r.Email__c,
    TaskBlock__r.Id, TaskBlock__r.Name,
    TaskBlock__r.Project__c, TaskBlock__r.StartDate__c,
    TaskBlock__r.EndDate__c, TaskBlock__r.Workload__c,
    TaskBlock__r.StartDate__c, TaskBlock__r.EndDate__c, TaskBlock__r.BaseStartDate__c, TaskBlock__r.BaseEndDate__c,
    TaskBlock__r.Workload__c, TaskBlock__r.AssignedWorkload__c,
    TaskBlock__r.Completion__c, TaskBlock__r.IsCompleted__c, TaskBlock__r.AssignedCost__c, TaskBlock__r.RealCost__c
    FROM Task__c
    WHERE TaskBlock__r.Project__c = :project.Id
    ORDER BY EndDate__c];

    List<TaskBlock__c> blocks = [SELECT Id, Name, Project__c, StartDate__c, EndDate__c,
    Completion__c, Workload__c, AssignedCost__c, RealCost__c
    FROM TaskBlock__c

```

```

WHERE Project__c = :project.Id
ORDER BY StartDate__c];

Map<Id, WrappedBlock> tBlockMap = new Map<Id, WrappedBlock>();
for(TaskBlock__c b: blocks) {
    tBlockMap.put(b.Id,
        new WrappedBlock(b));
}
for(Task__c t: tasks) {
    tBlockMap.get(t.taskBlock__r.Id).Tasks.add(t);
}
tBlocks = tBlockMap.values();
tBlocks.sort();

blockInput = false;

newTask = new Task__c();
newBlock = new TaskBlock__c(Project__c = project.Id);
}

// Method called when new task block input focus changes
public PageReference switchBlockInput() {
    blockInput = !blockInput;
    newBlock = new TaskBlock__c(Project__c = project.Id);
    return null;
}
// Method for inserting a new block and resetting tasks data
public PageReference addBlock() {
    insert newBlock;
    blockInput = false;
    resetTasksData();
    return null;
}
public PageReference deleteBlock() {
    delete newBlock;
    resetTasksData();
    return null;
}

// Method called when new task input focus changes
public PageReference switchTaskInput() {
    newTask = new Task__c(TaskBlock__c = newTBlockID);
    return null;
}
// Method for inserting a new task and resetting tasks data
public PageReference addTask() {
    insert newTask;
    newTBlockID = null;
    resetTasksData();
    return null;
}
// Method for deleting a task and resetting tasks data
public PageReference deleteTask() {
    delete newTask;
    resetTasksData();
    return null;
}

// Method called when new task assignment input focus changes
public PageReference switchTAInput() {
    newTaskAssignment = new TaskAssignment__c(Employee__c = newTAEmpID);
    return null;
}
// Method for inserting a new task assignment and resetting tasks data
public PageReference addTaskAssignment() {
    insert newTaskAssignment;
    newTAEmpID = null;
    resetResourcesData();
}

```



```

        return null;
    }
    // Method for deleting a task assignment and resetting tasks data
    public PageReference deleteTaskAssignment() {
        delete newTaskAssignment;
        resetResourcesData();
        return null;
    }

    // ***** Involved Resources *****

    // Method for retrieving all human resources related data
    public void resetResourcesData() {
        resources = [SELECT Id, Name, Calendar__c, Email__c,
                        (SELECT Id, Name, Task__r.Id, Task__r.Name, AssignedWorkload__c,
                        StartDate__c, EndDate__c,
                        BaseStartDate__c, BaseEndDate__c, Completion__c
                        FROM Employee__c.TaskAssignments__r
                        WHERE Task__r.TaskBlock__r.Project__c = :project.Id
                        ORDER BY StartDate__c)
                    FROM Employee__c
                    WHERE Id IN
                    (SELECT Employee__c FROM TaskAssignment__c WHERE
                    Task__r.TaskBlock__r.Project__c = :project.Id)
                    ORDER BY Name];

        newTaskAssignment = new TaskAssignment__c();
    }

    // ***** Emailing Interface *****

    public void sendProjectEmail() {
        // First, reserve email capacity for the current Apex transaction to ensure
        // that we won't exceed our daily email limits when sending email after
        // the current transaction is committed.
        Messaging.reserveSingleEmailCapacity(1);

        // Processes and actions involved in the Apex transaction occur next,
        // which conclude with sending a single email.

        // Now create a new single email message object
        // that will send out a single email to the addresses in the To, CC & BCC list.
        Messaging.SingleEmailMessage mail = new Messaging.SingleEmailMessage();

        // Strings to hold the email addresses to which you are sending the email.
        String[] toAddresses = new String[] {emailAddress};

        // Assign the addresses for the To and CC lists to the mail object.
        mail.setToAddresses(toAddresses);

        // Specify the address used when the recipients reply to the email.
        mail.setReplyTo(UserInfo.getUserEmail());

        // Specify the name used as the display name.
        mail.setSenderDisplayName(UserInfo.getFirstName() + ' ' + UserInfo.getLastName());

        // Specify the subject line for your email address.
        mail.setSubject(project.Name + ': ' + emailSubject);

        // Specify the text content of the email.
        mail.setPlainTextBody(emailText);

        // Send the email you have created.
        Messaging.sendEmail(new Messaging.SingleEmailMessage[] { mail });
    }

```

```

// ***** Wrapper Classes *****

// Wrapped date with information for completion charts
public class WrappedProjectDate {

    public Date ProjDate {get;set;}
    public Boolean IsCompleted {get;set;}
    public Integer ScheduledWorkload {get;set;}
    public Integer AssignedWorkload {get;set;}
    public Integer CompletedWorkload {get;set;}
    public Decimal BCWS {get;set;}
    public Decimal BCWP {get;set;}
    public Decimal ACWP {get;set;}

    public WrappedProjectDate(Date pDate, Boolean iC, Integer sw, Integer aw, Integer cw,
Decimal bS, Decimal bP, Decimal aP) {
        ProjDate = pDate;
        IsCompleted = iC;
        ScheduledWorkload = sw;
        AssignedWorkload = aw;
        CompletedWorkload = cw;
        BCWS = bS;
        BCWP = bP;
        ACWP = aP;
    }
}

public class tRow {
    public String name {get;set;}
    public String desc_PH {get;set;}
    public List<RowDetail> values {get;set;}
}

public class RowDetail {
    //actual variable name needed was "from", but it is keyword so we cant use that
    public String frm_PH {get;set;}
    public String to {get;set;}

    public String desc_PH {get;set;}
    public String label {get;set;}

    public String dep {get;set;}
    public String id {get;set;}

    /* Custom class Can be anything from below:
    ganttRed
    ganttGreen
    ganttOrange
    */
    public String customClass {get;set;}

    public void setToDate(String val) {
        to = setDate(val);
    }
    public void setFromDate(String val) {
        frm_PH = setDate(val);
    }
    //Date should be in special format as it is used in regular expression
    private String setDate(String val) {
        return '/Date('+val+')/';
    }
}

public class WrappedBlock implements Comparable {
    public List<Task__c> Tasks {get;set;}
}

```

```

public TaskBlock__c TaskBlock {get;set;}

public WrappedBlock(TaskBlock__c bTaskBlock) {
    Tasks = new List<Task__c>();
    TaskBlock = bTaskBlock;
}

public Integer compareTo(Object compareTo) {
    Integer result;
    WrappedBlock tBlock2 = (WrappedBlock) compareTo;

    if(TaskBlock.StartDate__c == null) {
        if(tBlock2.TaskBlock.StartDate__c == null) {
            if (TaskBlock.LastModifiedDate == tBlock2.TaskBlock.LastModifiedDate) {
                result = 0;
            } else if (TaskBlock.LastModifiedDate >
tBlock2.TaskBlock.LastModifiedDate) {
                result = 1;
            } else {
                result = -1;
            }
        } else {
            result = 1;
        }
    } else if(tBlock2.TaskBlock.StartDate__c == null) {
        result = -1;
    } else if (TaskBlock.StartDate__c == tBlock2.TaskBlock.StartDate__c) {
        if (TaskBlock.LastModifiedDate == tBlock2.TaskBlock.LastModifiedDate) {
            result = 0;
        } else if (TaskBlock.LastModifiedDate > tBlock2.TaskBlock.LastModifiedDate) {
            result = 1;
        } else {
            result = -1;
        }
    } else if (TaskBlock.StartDate__c > tBlock2.TaskBlock.StartDate__c) {
        result = 1;
    } else {
        result = -1;
    }
    return result;
}
}
}

```

BIBLIOGRAFÍA

- [1] D. Lock, Project Management, Aldershot: Gower Publishing Limited, 2007.
- [2] Project Management Institute, Guía de los Fundamentos de la Dirección de Proyectos (Guía del PMBOK) -- Quinta edición, Newtown Square, Pa.: Project Management Institute, Inc., 2013.
- [3] Project Management Institute, «Learn More About Who PMI is and What We Do,» 2014. [En línea]. Available: <http://www.pmi.org/About-Us.aspx>. [Último acceso: 25 Abril 2014].
- [4] Project Management Institute, Inc., «PMI FS-PMBOK-2013,» American National Standards Institute, 2013.
- [5] Axelos, «Projects in Controlled Environments, PRINCE2 Introduction,» 2014. [En línea]. Available: <http://www.axelos.com/Project-Management-PRINCE2/>. [Último acceso: 25 Abril 2014].
- [6] A. Murray, *PRINCE2 in one thousand words*, Londres: The Stationery Office, 2011.
- [7] Axelos, «ITIL and IT Service Management,» ITIL & IT Service Management Bookshop, 2014. [En línea]. Available: <http://www.itil.org.uk/>. [Último acceso: 25 Abril 2014].
- [8] OSIATIS S.A., «Fundamentos de la Gestión TI - ITIL - ¿Qué es ITIL?,» 2014. [En línea]. Available: http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/fundamentos_de_la_gestion_TI/que_es_ITIL/que_es_ITIL.php. [Último acceso: 25 Abril 2014].
- [9] W. F. Abba, «How Earned Value Got to Prime Time: A Short Look Back and Glance Ahead,» *The Measurable News*, 2001.
- [10] College of Performance Management, «What Is EVM?,» 2014. [En línea]. Available: <http://www.mycpm.org/what-is-evm/>. [Último acceso: 25 Abril 2014].
- [11] Defense Acquisition University, *Earned Value Management 'Gold Card'*, 2010.
- [12] T. Raz, R. Barnes y D. Dov, «A Critical Look at Critical Chain Project Management,» *Project Management Journal*, nº Dec 2003, pp. 24-32, 2003.

- [13] L. P. Leach, «Critical Chain Project Management Improves Project Performance,» Project Management Institute Inc., Idaho Falls, 1999.
- [14] J. Palacio, Flexibilidad con Scrum, 2007.
- [15] K. Schwaber y J. Sutherland, Scrum Guide, 2013.
- [16] E. Burgos, «Project Management = BASECAMP,» 13 Enero 2011. [En línea]. Available: <http://www.enriqueburgos.com/2011/01/project-management-basecamp/>. [Último acceso: 25 Abril 2014].
- [17] J. J. Velasco, «Smartsheet: gestión profesional de proyectos en la nube usando Google Drive,» 24 Noviembre 2012. [En línea]. Available: <http://bitelia.com/2012/11/smartsheet-gestion-proyectos-nube>. [Último acceso: 25 Abril 2014].
- [18] M. Benioff y C. Adler, Behind the Cloud: The Untold Story of How Salesforce.com Went from Idea to Billion-Dollar Company and Revolutionized an Industry, Jossey-Bass, 2009.
- [19] N. Kolakowski, «Salesforce CEO Marc Benioff Talks Cloud Computing, Twitter,» *eWeek*, 2009.
- [20] M. Rouse, «What is model-view-controller (MVC)? - Definition from WhatIs.com,» Marzo 2011. [En línea]. Available: <http://whatis.techtarget.com/definition/model-view-controller-MVC>. [Último acceso: 16 Mayo 2014].
- [21] Salesforce, «Getting Started with Encrypted Custom Fields,» 2008. [En línea]. Available: <http://help.salesforce.com/servlet/servlet.FileDownload?file=015300000034WplAAE>. [Último acceso: 24 Mayo 2014].
- [22] A. Shaw, «FullCalendar,» [En línea]. Available: <http://arshaw.com/fullcalendar/>. [Último acceso: 2014 Junio 5].
- [23] C. Sechelski, «Create a Calendar View in Salesforce.com,» [En línea]. Available: <http://www.codebycody.com/2013/06/create-calendar-view-in-salesforcecom.html>. [Último acceso: 2014 Junio 5].
- [24] Google, «Google Charts - Google Developers,» [En línea]. Available:

<https://developers.google.com/chart/>. [Último acceso: 2014 Junio 05].

- [25] mbielanczuk, «About jQuery Gantt Chart,» [En línea]. Available: <http://mbielanczuk.com/jquery-gantt/>. [Último acceso: 2014 Junio 5].
- [26] J. Zaa, «Gantt Chart in Salesforce using JQuery and JSON,» [En línea]. Available: <http://www.shivasoft.in/blog/salesforce/gantt-chart-in-salesforce-using-jquery-and-json/>. [Último acceso: 2014 Junio 5].
- [27] User Interface Design GmbH, «AttrakDiff,» [En línea]. Available: <http://attrakdiff.de/index-en.html>. [Último acceso: 2014 Junio 6].
- [28] L. F. Sánchez-Arias y L. Solarte-Pazos, «El cuerpo de conocimientos del Project Management Institute-PMBOK® Guide, y las especificidades de la gestión de proyectos. Una revisión crítica,» *Innovar*, vol. 20, nº 37, pp. 89-100, 2010.
- [29] P. W. G. Morris, *The Management of Projects*, London: Thomas Telford Services Limited, 1994.
- [30] Departamento de Defensa - Estados Unidos, «Earned Value Management (EVM) Division,» 2014. [En línea]. Available: <http://www.acq.osd.mil/evm/>. [Último acceso: 25 Abril 2014].

Este documento esta firmado por



Firmante	CN=tfgm.fi.upm.es, OU=CCFI, O=Facultad de Informatica - UPM, C=ES
Fecha/Hora	Fri Jun 06 23:47:44 CEST 2014
Emisor del Certificado	EMAILADDRESS=camanager@fi.upm.es, CN=CA Facultad de Informatica, O=Facultad de Informatica - UPM, C=ES
Numero de Serie	630
Metodo	urn:adobe.com:Adobe.PPKLite:adbe.pkcs7.sha1 (Adobe Signature)